Calhoun: The NPS Institutional Archive
DSpace Repository

Theses and Dissertations 1. Thesis and Dissertation Collection, all items

1983-09

A linear approximation of the source position using multiple MAD

Bock, Wolf-Hubertus

Monterey, California. Naval Postgraduate School

http://hdl.handle.net/10945/19868

Downloaded from NPS Archive: Calhoun

Calhoun is the Naval Postgraduate School's public access digital repository for research materials and institutional publications created by the NPS community. Calhoun is named for Professor of Mathematics Guy K. Calhoun, NPS's first appointed -- and published -- scholarly author.

Dudley Knox Library / Naval Postgraduate School
411 Dyer Road / 1 University Circle
Monterey, California USA 93943

http://www.nps.edu/library

Dudley Knox Library, NPS
Monterey, CA 93943

NAVAL POSTGRADUATE SCHOOL
Monterey, California

THESIS

A LINEAR APPROXIMATION OF THE SOURCE POSITION USING MULTIPLE MAD

by

Wolf-Hubertus Bock

September 1983

Thesis Advisor: Andrew R. Ochadlick, jr.

Approved for public release; distribution unlimited.

T210144

Dudley Knox Library, N
Monterey, CA

UNCLASSIFIED
SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) A Linear Approximation of the Source Position Using Multiple MAD		5. TYPE OF REPORT & PERIOD COVERED Master's Thesis September 1983
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Wolf-Hubertus Bock		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS Naval Postgraduate School Monterey, California 93943		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS Naval Postgraduate School Monterey, California 93943		12. REPORT DATE September 1983
		13. NUMBER OF PAGES 61
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS. (of this report)
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

MAD localization, Multiple MAD, Non-acoustic localization

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

For certain assumptions, an analysis of multiple MAD signals results in a reasonable estimate for the localization of a target relative to the MAD platform. This is achieved by using selective approximations to linearize an initially nonlinear problem. The simulation ignores noise and requires an estimate of the magnitude of the target magnetic moment components. Results indicate that the best localization estimates are

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE
S/N 0102-LF-014-6601

UNCLASSIFIED
SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

achieved when the platform is on cardinal headings, and when the target moment has a strong vertical component.

Approved for public release; distribution unlimited.

A Linear Approximation of the Source Position Using Multiple MAD

by

Wolf-Hubertus Bock
Lieutenant, United States Navy
B.A., Rice University, 1975

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN SYSTEMS TECHNOLOGY
(ANTISUBMARINE WARFARE)

from the

NAVAL POSTGRADUATE SCHOOL
September 1983

no.
B5825
c.1

ABSTRACT

For certain assumptions, an analysis of multiple MAD signals results in a reasonable estimate for the localization of a target relative to the MAD platform. This is achieved by using selective approximations to linearize an initially nonlinear problem. The simulation ignores noise and requires an estimate of the magnitude of the target magnetic moment components. Results indicate that the best localization estimates are achieved when the platform is on cardinal headings, and when the target moment has a strong vertical component.

TABLE OF CONTENTS

I. INTRODUCTION 9

A. BACKGROUND 9

B. APPROACH 10

II. THEORY 14

III. NATURE OF THE PROBLEM 18

IV. DISCUSSION 23

A. GENERAL 23

B. HEADING CHANGES 29

C. TARGET DIPOLE MOMENT CHANGES 30

D. CHANGES IN SENSOR SPACING 30

E. ALTITUDE CHANGES 31

F. NON-OVERHEAD PASSES 31

G. AVERAGES 31

H. IMPORTANCE OF CONSISTENCY 41

I. IMPLEMENTATION 41

V. CONCLUSION 42

APPENDIX A: COMPUTER PROGRAM FOR PLOT OUTPUT 44

APPENDIX B: COMPUTER PROGRAM FOR NUMERICAL OUTPUT . . . 50

APPENDIX C: EXAMPLE OUTPUT OF APPENDIX B 55

APPENDIX D: CRAMER'S RULE 57

A. GENERAL 57

B. EXAMPLE 57

C. SUBROUTINE "CRAMER'S RULE" CHECK 58

LIST OF REFERENCES 59

BIBLIOGRAPHY 60

INITIAL DISTRIBUTION LIST 61

LIST OF FIGURES

1.1 Spatial Relationship of Sensor Positions 12
1.2 Spatial Relationship of Platform and Target . . 13
2.1 Vector Relationship of $\vec{B}_t$, $\vec{b}_d$, and $\vec{B}_e$ 14
4.1 Initial Run, x position 24
4.2 Initial Run, y position 25
4.3 Initial Run, z position 26
4.4 Initial Run, corrected y position 27
4.5 Platform Heading 60° Magnetic 33
4.6 Platform Heading 120° Magnetic 34
4.7 Platform Heading 240° Magnetic 35
4.8 Platform Heading 300° Magnetic 36
4.9 $\vec{m}$ Vertically Up 37
4.10 $\vec{m}$ Horizontal and Oriented to Magnetic North . . 38
4.11 $\vec{m}$ Vert. Comp. 50° Down, $\vec{m}$ Hor. Comp. 30° Magnetic 39
4.12 Target at 45° depres. angle rel. to platform at CPA 40

TABLE OF SYMBOLS

$\vec{b}_d$	magnetic field vector of the target dipole
$\vec{B}_e$	earth's magnetic field vector
$\vec{B}_t$	total magnetic field vector
$\hat{e}$	earth's magnetic field unit vector
ex,ey,ez	components of $\hat{e}$ in cartesian coordinates
$\vec{r}$	vector from the target to the platform
r	scalar magnitude of $\vec{r}$

I. INTRODUCTION

A. BACKGROUND

Ever since the operational introduction of the MAD (Magnetic Anomaly Detection) system in 1944, it has been a single sensor system limited to indicating detection only. The short ranges of the early versions allowed a general localization if a signal was detected. The use of multiple sensors should allow for a good estimate of the target's location relative to the sensor platform, assuming that the arrangement of the sensors is such that the received signals are linearly independent of one another. Previous attempts to solve this problem have in general focused on the use of only two sensors, but did not develop a usable solution to operational completion. Work has been done on this topic by Wynn et al., [Ref. 1], using a superconducting gradiometer array. The measured signals in their work were subjected to a novel signal processing technique which, in the laboratory, was used to invert the dipole field equation to determine both the position and the moment vectors of a dipole signal source. This thesis approaches the same problem by simulating five standard total field measurements. As shown in Figure 1.1 , four magnetometers moving in parallel paths are used to obtain these measurements. The advantage of this approach compared to that of Wynn et al. is that it utilizes existing technology in conjunction with approximations, delineated in Chapter 3, in order to produce a usable result.

B. APPROACH

The problem of localizing a target from any platform involves at least three unknowns: r_x, r_y, and r_z, the components of the position vector from the platform to the target. In the case of using MAD, three additional unknowns m_x, m_y, and m_z, the components of the target's magnetic moment, are also involved. The result is that there are six unknowns that must be solved for in order to localize a target using MAD. Since a normal sized magnetic target can be approximated by the magnetic field of a dipole [Ref. 2], it was decided to use the dipole equation as described in [Ref. 3] to generate the signals used in this simulation. Equation 1.1 describes the MAD signal, and will be developed from the dipole equation in chapter 2. Equation 1.1 is non-

$$S(\vec{r}) = -(m_x e_x + m_y e_y + m_z e_z)/r^3 + 3(m_x r_x + m_y r_y + m_z r_z)(r_x e_x + r_y e_y + r_z e_z)/r^5 \qquad \text{(eqn 1.1)}$$

linear in the unknowns. The position vector in equation 1.1 is defined as going from the target to the platform; this is a consequence of using the dipole equation, and will be explained later.

The cartesian coordinates are defined in Figure 1.2 . The z axis is in the direction of platform motion, and the y axis is vertical down. The components of the earth's field unit vector, e_x, e_y, and e_z, are assumed to be known quantities. Since it is very difficult to explicitly solve a set of non-linear equations in six unknowns, the problem will be simplified by making assumptions and dividing it into two special cases. The first case is a trivial one in which the target's position is assumed known. For this case, equation 1.1 is linear in the three unknowns m_x, m_y, and m_z. If the three measurements are linearly independent, then Cramer's Rule, [Ref. 4], may be used to solve for these unknowns. The

second case is the more difficult one, and it is the problem investigated in this thesis. In this case, the assumption is made that the target's magnetic moment is known. That reduces equation 1.1 into a non-linear equation in only three unknowns, r_x, r_y, and r_z. If this simplified version of equation 1.1 could be manipulated into a linear form by using several sensors and appropriate combinations of signals, the position of the target could easily be solved for.

This thesis will develop an approximate solution to this problem by using combinations of the total field signals of an idealized multiple MAD system.

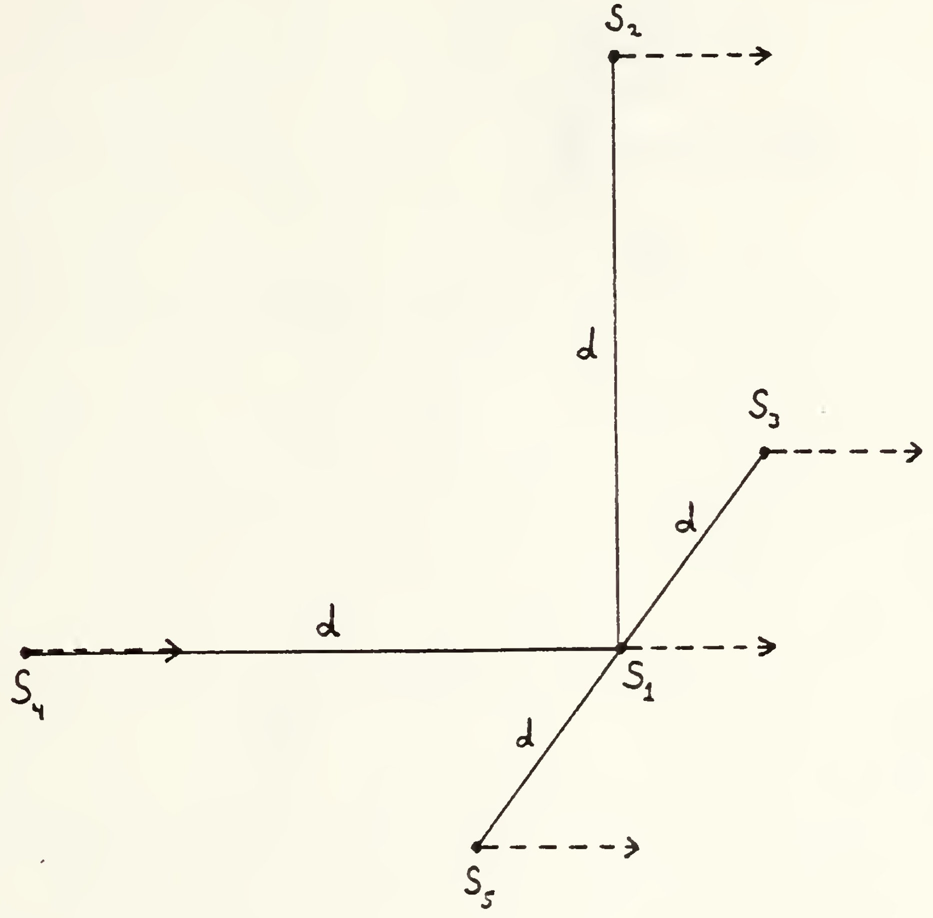


Figure 1.1 Spatial Relationship of Sensor Positions.

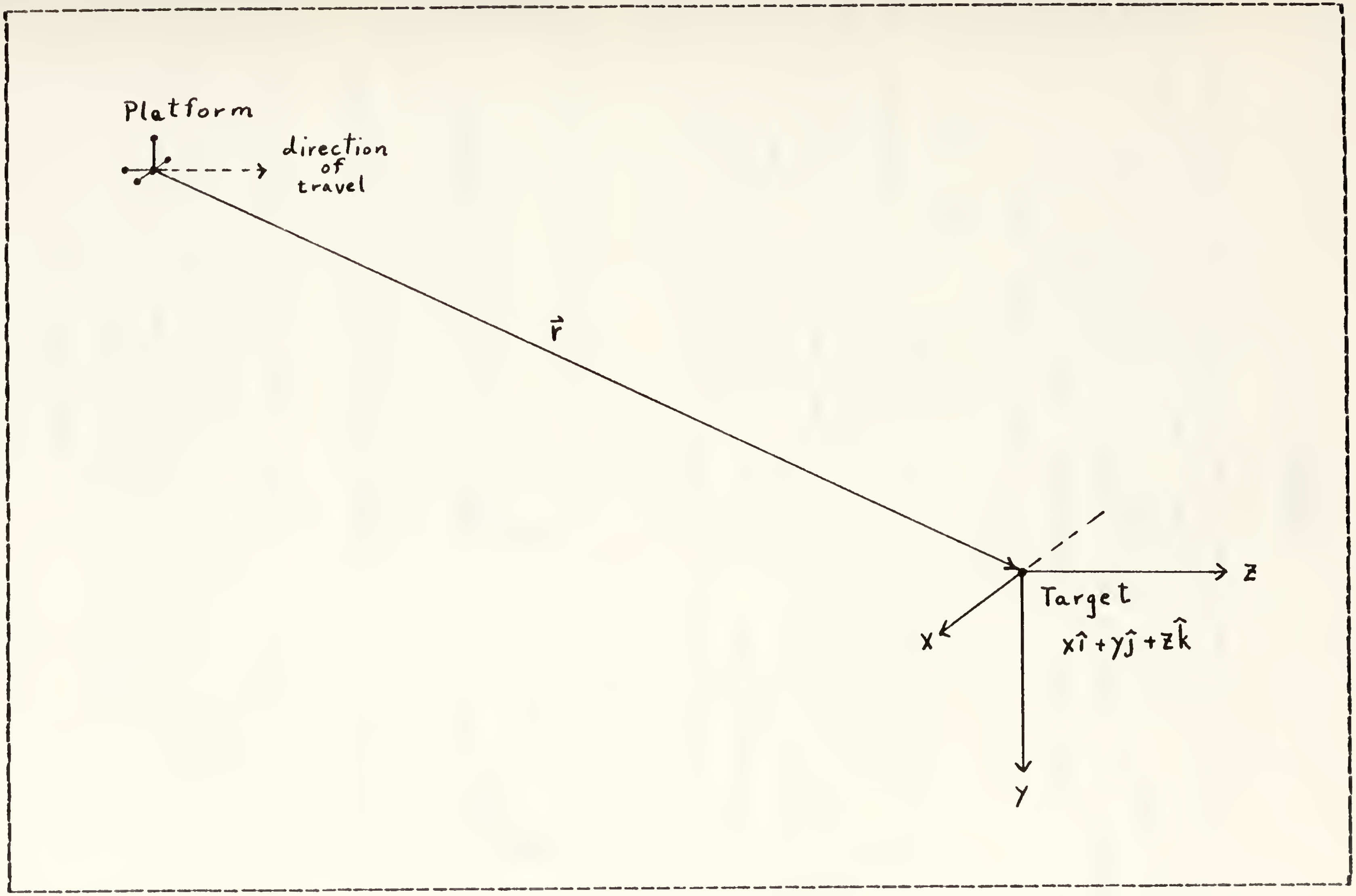


Figure 1.2 Spatial Relationship of Platform and Target.

II. THEORY

The total field magnetometer measures the magnitude of the vector sum of the ambient magnetic field of the earth and the magnetic field of a magnetic dipole moment. The field of the dipole will used to represent the magnetic field of the target. The magnitude of the vector sum is called the total field as given by

$$\vec{B}_t = |\vec{B}_e + \vec{b}_\alpha|$$

$\vec{B}_e$ represents the earth's magnetic field vector and $\vec{b}_\alpha$ represents the field due to a magnetic dipole. In the general case of interest here, the magnitude of the dipole field is much less than the magnitude of the earth's field. Under certain conditions, a useful approximation to the total field may be derived from the geometry shown in Figure 2.1 .

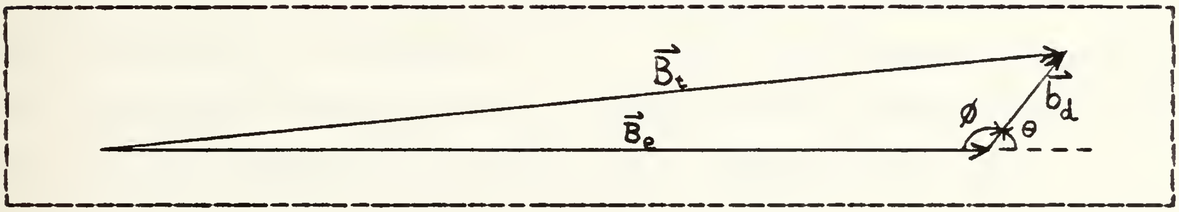


Figure 2.1 Vector Relationship of $\vec{B}_t$, $\vec{b}_\alpha$, and $\vec{B}_e$.

In this figure a very small vector is added to a very large one. The vectors of Figure 2.1 are not to scale. $\vec{b}_\alpha$ is to be considered several orders of magnitude smaller than $\vec{B}_e$. The application of basic geometric formulas for the sides of a triangle and the angle between sides yields

$$B_t{}^2 = B_e{}^2 + b_\alpha{}^2 - 2b_\alpha B_e\cos\phi$$

for the geometry of Figure 2.1 . In terms of θ this equation becomes

$$B_t{}^2 = B_e{}^2 + b_d{}^2 + 2b_dB_e\cos\theta$$

The $B_e{}^2$ term can be factored out of the right hand side of the equation to produce

$$B_t{}^2 = B_e{}^2\{1 + (b_d/B_e)^2 + 2(b_d/B_e)\cos\theta\}$$

Taking the square root yields equation 2.1 .

$$B_t = B_e\{[1 + (b_d/B_e)^2 + 2(b_d/B_e)\cos\theta]^{1/2}\} \qquad \text{(eqn 2.1)}$$

Since by assumption b_d is much less than B_e, the term $(b_d/B_e)^2$ is much less than $2(b_d/B_e)\cos\theta$ and can be considered negligible. This reduces equation 2.1 to

$$B_t = B_e\{[1 + 2(b_d/B_e)\cos\theta]^{1/2}\} \qquad \text{(eqn 2.2)}$$

The general expansion formula given by

$$(1+x)^n = 1 + nx + n(n-1)x^2/2! + \text{Higher Order Terms}$$

can be applied to equation 2.1 . For the case here, $n=1/2$, and $x=(b_d/B_e)^2+2(b_d/B_e)\cos\theta$. Substituting these values into the general expansion formula gives equation 2.1 as

$$\begin{aligned} B_t &= B_e\{1 + .5[(b_d/B_e)^2 + 2(b_d/B_e)\cos\theta] \\ &\quad - 1/8[(b_d/B_e)^2 + (2b_d/B_e)\cos\theta]^2 + \text{H.O.T.}\} \end{aligned}$$

Squaring the appropriate term and multiplying through by the coefficients results in

$$\begin{aligned} B_t &= B_e\{1 + .5[(b_d/B_e)^2] + (b_d/B_e)\cos\theta - 1/8[b_d/B_e]^4 \\ &\quad - 4/8[b_d/B_e]^3\cos\theta - 4/8[b_d/B_e]^2\cos^2\theta + \text{H.O.T.}\} \end{aligned}$$

The cubed and higher order terms will be considered to be negligible. Dropping those terms and rearranging the remaining ones yields the form

$$B_t = B_e\{1 + (b_d/B_e)\cos\theta - .5(b_d/B_e)^2\cos^2\theta + .5(b_d/B_e)^2\}$$

Using the relation $\sin^2\theta + \cos^2\theta = 1$, this reduces to

$$B_t = B_e\{1 + (b_d/B_e)\cos\theta + .5(b_d/B_e)^2\sin^2\theta\}$$

After multiplying through by B_e, the second term can be written in terms of the dot product $\vec{b_d}\bullet\vec{B_e}/B_e$. This is reduced to $\vec{b_d}\bullet\hat{e}$ if the unit vector $\hat{e}$ is defined as $\vec{B_e}/B_e$, and the resulting equation is

$$B_t = B_e + \vec{b_d}\bullet\hat{e} + .5(b_d/B_e)^2\sin^2\theta$$

Since b_d is on the order of 1nT while B_e is on the order of 50,000 nT, the error term involving the sine is negligible. For the stated assumptions, equation 2.1 can be represented by

$$B_t = B_e + \vec{b_d}\bullet\hat{e} \qquad \text{(eqn 2.3)}$$

In this thesis, B_e will be assumed constant. In addition, this simulation will assume that no noise exists. Thus, the total field magnetometer will measure the anomaly in the earth's field defined by

$$S(\vec{r}) = \vec{b_d}\bullet\hat{e}$$

In this expression, $\vec{b_d}$ represents the magnetic field of a dipole. $\vec{b_d}$ is defined [Ref. 3] by the equation

$$\vec{b_d}(\vec{r}) = -\vec{m}/r^3 + 3(\vec{m}\bullet\vec{r})\vec{r}/r^5 \qquad \text{(eqn 2.4)}$$

$\vec{m}$ is the magnetic dipole moment in units of nTft³. $\vec{r}$ is the position vector to the field point and is measured in feet. If b_d is much less than B_e, then equation 2.3 and equation 2.4 result in the equation

$$S(\vec{r})=-(\vec{m}\bullet\hat{e})/r^3+3(\vec{m}\bullet\vec{r})(\vec{r}\bullet\hat{e})/r^5 \qquad \text{(eqn 2.5)}$$

This equation is an alternative basis for the derivation of the Anderson's Functions, found in [Ref. 5]. The Anderson Functions are a commonly used approximation for the signal received by a MAD sensor. The computer simulation in this thesis is based on equation 2.5 directly, since it is in a representation which is easier to use than the Anderson's function representation of a MAD signal.

III. NATURE OF THE PROBLEM

This thesis will develop and demonstrate an approximation that generates a reasonable solution of the target's position relative to a platform bearing several "sensors" to "measure" the magnetic signal as the platform makes a straight line encounter near the target. The assumptions are as follows:

1.) The dipole equation is a valid representation of the magnetic field of the target.

2.) b_{α} is several orders of magnitude smaller than B_e.

3.) Measurements are done using only straight line encounters.

4.) The target's magnetic dipole moment is a known vector.

5.) The earth magnetic field vector is known.

6.) The gradient of the earth's field is zero.

7.) No magnetic noise of any kind (environment, sensor, etc) exists.

8.) The measurements are oriented along four lines which are oriented with respect to the flight path as shown in Figure 1.1 .

Since the dipole equation is written with the moment at the origin of a coordinate system, the calculations of the signal field values used in the simulation are made with the target located at the origin. As will be seen, the program performs a coordinate transformation in order to produce the output of target position relative to the platform. After this coordinate transformation, the origin is located at the sensor platform and moves with it.

The five measurements are made along four separate flight paths, so only four different "sensors" are involved.

Sensor 4 could be replaced by inducing a time delay in the measurement of sensor 1. The signals generated at these measurement points are measured at each time step. The position vectors can all be described relative to the origin, located at the target, in terms of the position vector of sensor 1. The measured signals are represented by the following equations:

$$S(\vec{r1})=-\{(m_xe_x+m_ye_y+m_ze_z)(r1)^2-3(m_xx+m_yy+m_zz)(xe_x+ye_y+ze_z)\}/r1^5$$
$$S(\vec{r2})=-\{(m_xe_x+m_ye_y+m_ze_z)(r2)^2-3(m_xx+m_y[y-d]+m_zz)(xe_x+[y-d]e_y+ze_z)\}/r2^5$$
$$S(\vec{r3})=-\{(m_xe_x+m_ye_y+m_ze_z)(r3)^2-3(m_x[x-d]+m_yy+m_zz)([x-d]e_x+ye_y+ze_z)\}/r3^5$$
$$S(\vec{r4})=-\{(m_xe_x+m_ye_y+m_ze_z)(r4)^2-3(m_xx+m_yy+m_z[z-d])(xe_x+ye_y+[z-d]e_z)\}/r4^5$$
$$S(\vec{r5})=-\{(m_xe_x+m_ye_y+m_ze_z)(r5)^2-3(m_x[x+d]+m_yy+m_zz)([x+d]e_x+ye_y+ze_z)\}/r5^5$$

For convenience, the following dummy variables are used in the computer program and are defined as:

$G = m_xe_x+m_ye_y+m_ze_z$
$V1 = 3m_xe_x - G$
$V2 = 3(m_xe_y + m_ye_x)$
$V3 = 3m_ye_y - G$
$V4 = 3(m_ye_z + m_ze_y)$
$V5 = 3m_ze_z - G$
$V6 = 3(m_ze_x - m_xe_z)$
$S1 = S(\vec{r1})$
$S2 = S(\vec{r2})$
$S3 = S(\vec{r3})$
$S4 = S(\vec{r4})$
$S5 = S(\vec{r5})$

The use of these variables reduces the above equations to the following form, which are the defining equations for the signals in the simulation.

$$S1=(V1x^2+V2xy+V3y^2+V4yz+V5z^2+V6xz)/r1^5$$

$$S2=(V1x^2+V2xy-V2xd+V3y^2-2V3yd+V3d^2+V4yz-V4dz+V5z^2+V6xz)/r2^5$$

$$S3=(V1x^2-2V1xd+V1d^2+V2xy-V2dy+V3y^2+V4yz+V5z^2+V6xz-V6dz)/r3^5$$

$$S4=(V1x^2+V2xy+V3y^2+V4yz-V4yd+V5z^2-2V5zd+V5d^2+V6xz-V6xd)/r4^5$$

$$S5=(V1x^2+2V1xd+V1d^2+V2xy+V2dy+V3y^2+V4yz+V5z^2+V6xz+V6dz)/r5^5$$

The unknowns of interest are x, y, and z, which are related to the position vectors as follows:

$$\vec{r1} = x\hat{i} + y\hat{j} + z\hat{k} \qquad \text{(eqn 3.1)}$$

$$\vec{r2} = x\hat{i} + (y-d)\hat{j} + z\hat{k} \qquad \text{(eqn 3.2)}$$

$$\vec{r3} = (x-d)\hat{i} + y\hat{j} + z\hat{k} \qquad \text{(eqn 3.3)}$$

$$\vec{r4} = x\hat{i} + y\hat{j} + (z-d)\hat{k} \qquad \text{(eqn 3.4)}$$

$$\vec{r5} = (x+d)\hat{i} + y\hat{j} + z\hat{k} \qquad \text{(eqn 3.5)}$$

To linearize the problem it will be assumed that the denominators of the equations for S1, S2, S3, S4, and S5 given above are all equal to r, i.e.,

$$r = r1 = r2 = r3 = r4 = r5$$

The equations for S1, S2, S3, S4, and S5 with the denominators equal can then be subtracted from one another to yield the following:

$$S2-S3=\{(V2-2V1)x+(2V3-V2)y+(V4-V6)z+(V1-V3)d\}(-d/r^5)$$

$$S2-S5=\{(V2+2V1)x+(2V3+V2)y+(V4+V5)z+(V1-V3)d\}(-d/r^5)$$

$$S3-S1=\{2V1x+V2y+V6z-V1d\}(-d/r^5)$$

$$S5-S1=\{-2V1x-V2y-V6z-V1d\}(-d/r^5)$$

$$S3-S4=\{(2V1-V6)x+(V2-V4)y+(V6-2V5)z+(V5-V1)d\}(-d/r^5)$$

$$S5-S4=\{-(2V1+V6)x-(V2+V4)y-(V6+2V5)z+(V5-V1)d\}(-d/r^5)$$

Since the denominators are assumed to be identical, if these six equations are divided in pairs the coefficients involving the non-linear r^5 term and the quantity d are canceled out. This results in the following equations that define the new variables A1, A2, and A3 as

$$A1 = (S2-S3)/(S2-S5) \qquad \text{(eqn 3.6)}$$

$$A2 = (S3-S1)/(S5-S1) \qquad \text{(eqn 3.7)}$$

$$A3 = (S3-S4)/(S5-S4) \qquad \text{(eqn 3.8)}$$

Defining the following variables for convenience,

$C1 = A1(V2+2V1) - (V2-2V1)$

$C2 = A1(2V3+V2) - (2V3-V2)$

$C3 = A1(V4+V6) - (V4-V6)$

$K1 = [(V1-V3) - A1(V1-V3)]d$

$C4 = A2(2V1) + 2V1$

$C5 = A2(V2) + V2$

$C6 = A2(V6) + V6$

$K2 = [V1 - A2(V1)]d$

$C7 = A3(V6+2V1) - (V6-2V1)$

$C8 = A3(V4+V2) - (V4-V2)$

$C9 = A3(2V5+V6) - (2V5-V6)$

$K3 = [(V1-V5) - A3(V1-V5)]d$

equations 3.6 - 3.8 can be written in the following form as

$C1x + C2y + C3z = K1$ (eqn 3.9)

$C4x + C5y + C6z = K2$ (eqn 3.10)

$C7x + C8y + C9z = K3$ (eqn 3.11)

These three equations are linear in terms of x, y, and z. In addition, they are linearly independent as shown by using the program in Appendix B. A sample output demonstrating the linear independence is shown in Appendix C. Since they are linearly independent, Cramer's Rule can be used to solve equations 3.9 - 3.11 for x, y, and z. The computer program in Appendix A was written in terms of these equations. The program assumes that the magnitude of the target's magnetic moment is 5x10^8 nTft^3. A representative moment was selected by using the values listed by Fromm in [Ref. 2], who lists a moment of 10^8 to 2x10^8 cgs units for a submarine. One cgs unit is approximately 3.35 nTft^3. The value used in the program is a rounded out average of Fromm's values. Computations are done using the standard convention with the target dipole at the origin. However, as indicated earlier,

the output is corrected by a coordinate transformation so that the platform is at the origin, and the target position is given with respect to the platform.

IV. DISCUSSION

A. GENERAL

The program in Appendix B was written to check whether the results generated by the approximation equations were reasonably close to the actual coordinates of the target. The program was modified into the form shown in Appendix A to graphically provide a comparison between the calculated and the actual position coordinates. Initial parameters for the knowns in the equations were a platform heading of 30° magnetic, an earth field vector of 70° down from the horizontal, and a target dipole moment vector with a vertical component 50° down from the horizontal and a horizontal component oriented 355° from magnetic north. These initial parameters have no special significance, and merely represent a convenient starting point. As discussed earlier, the magnitude of the target dipole moment was set at the value of $5X10^8$ $nTft^3$. The spacing between measurement positions, d, was taken as 50 feet, and the platform passed directly overhead the target at 1000 feet. This and all subsequent runs were started at 5000 feet prior to CPA (closest point of approach) with the platform advancing 50 feet per time step. One position calculation was completed at each time step, the simulation ceasing when the platform was 5000 feet past CPA. For interpretation purposes, calculated target positions are shown as the sensor platform moves 10,000 feet past the target in a straight line encounter, with CPA occurring at the 5000 foot point. Neither the CPA nor the target's location are in any way considered known in the localization process. However, as the area of interest is the behavior of the simulation close to CPA, the closest

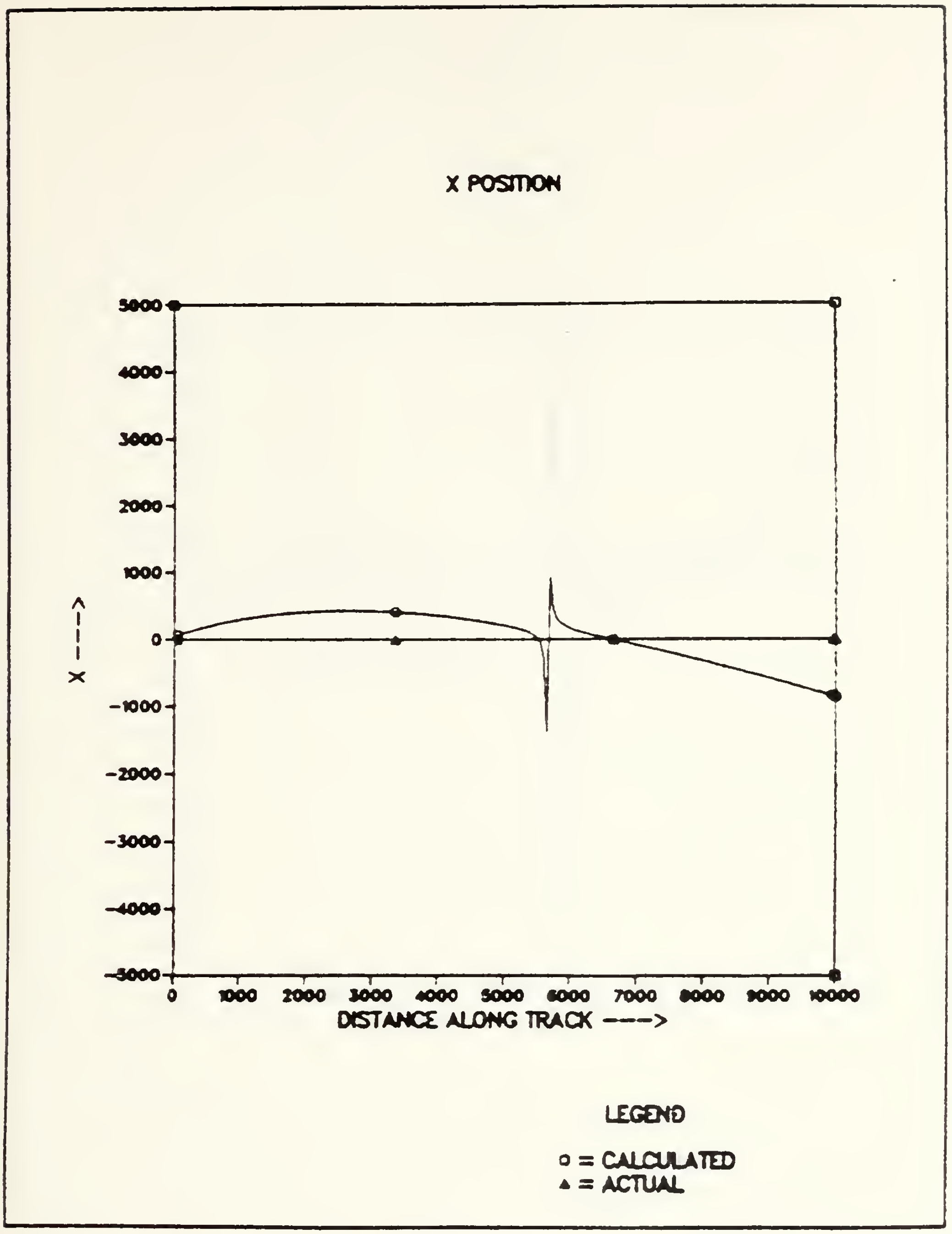


Figure 4.1 Initial Run, x position.

point of approach is always centered in the following figures.

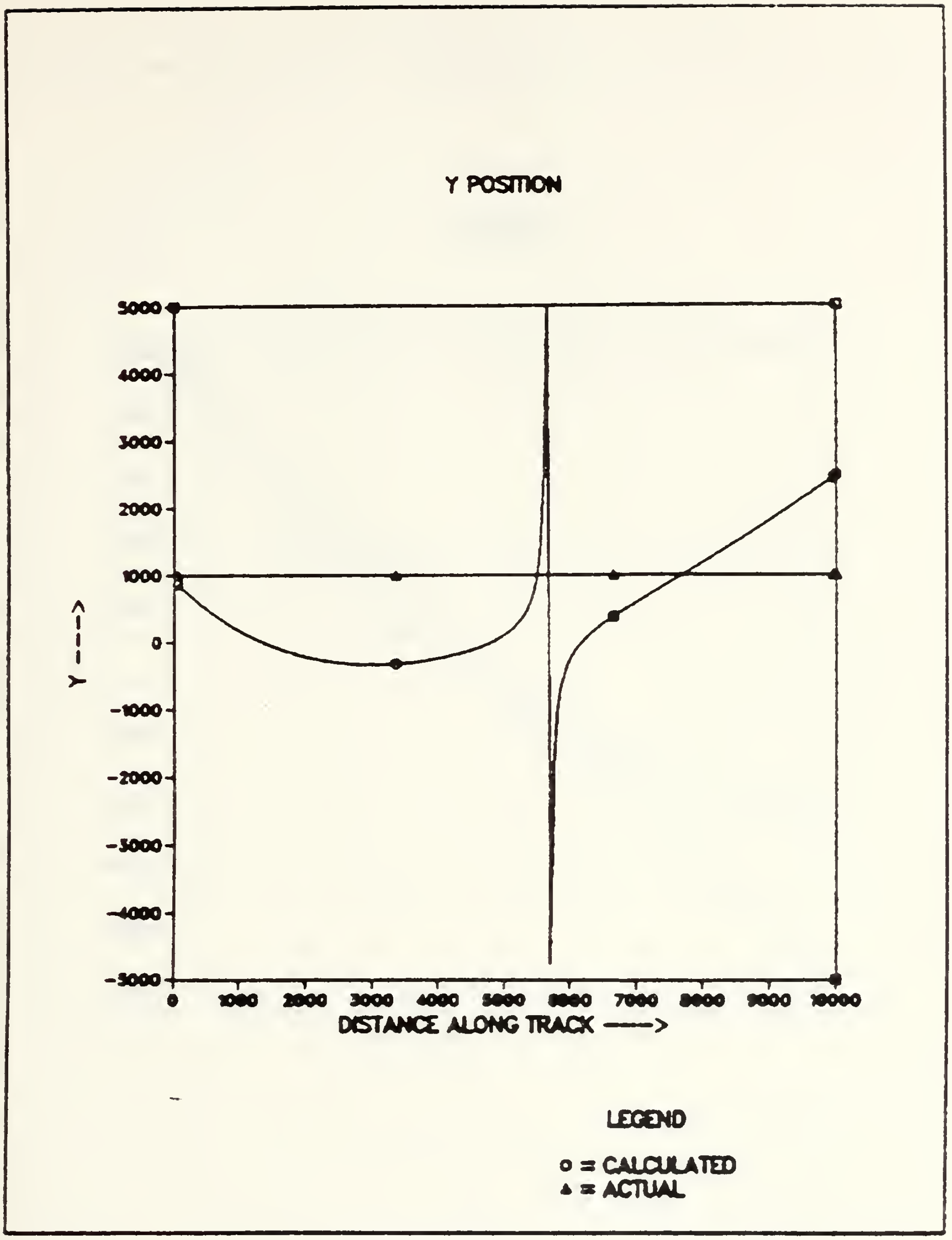


Figure 4.2 Initial Run, y position.

From the orientation of the coordinate system shown in Figure 1.2 , z is in the direction of flight, y is straight down, and x is off the right side of the platform. Since

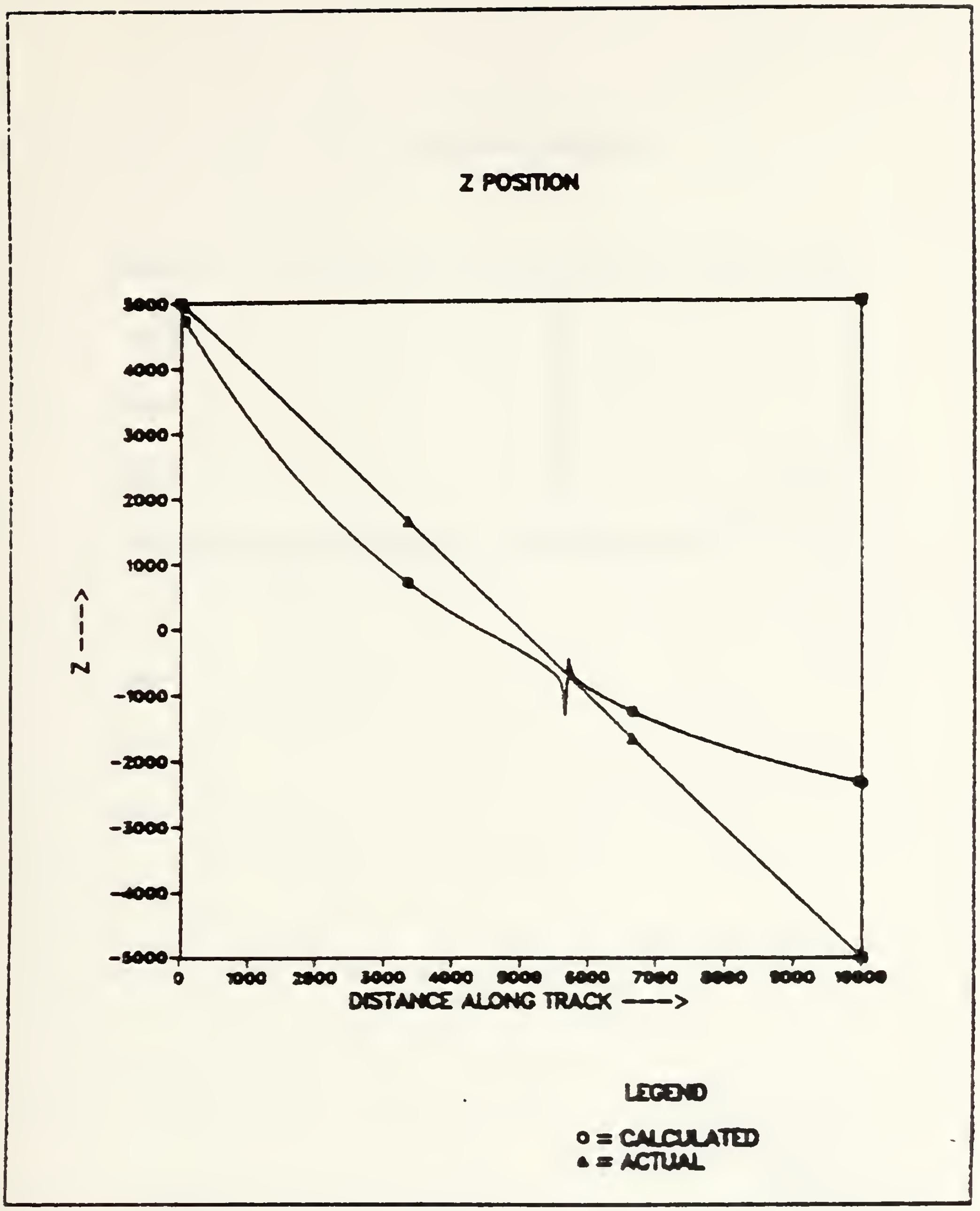


Figure 4.3 Initial Run, z position.

this thesis investigates straight line encounters only, the x(actual) and y(actual) coordinates will remain constant throughout any run for the coordinate orientation used. The

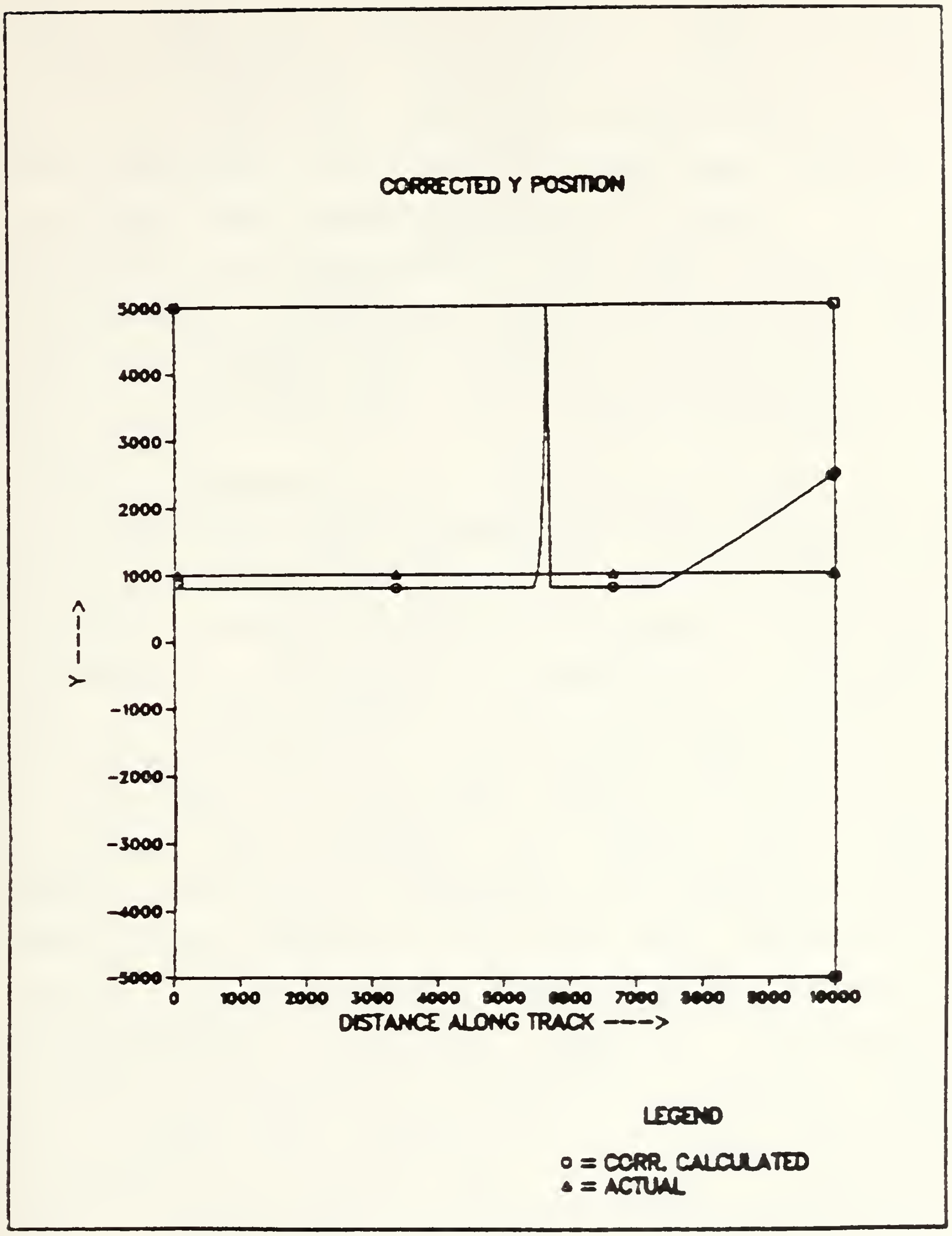


Figure 4.4 Initial Run, corrected y position.

five measurements will therefore be made with their respective x and y coordinates as constants. The z coordinate will vary. This means that for the initial pass described,

x(actual) would be constant at zero, y(actual) would be constant at 1000 feet, while z(actual) would progress from 5000 feet to -5000 feet. Figures 4.1 - 4.3 show a comparison of the actual and calculated target positions as the platform completes the initial run described earlier. It will be noted that the calculated and actual x, y, and z positions are almost identical at the start of the run. This is a coincidence. "Crossover" points such as this occur for all three coordinates simultaneously during a run, falling closer or further away from CPA, depending upon the choice of parameters.

In this initial run, it is obvious that the "worst" estimates occur for the y coordinate. This is less significant than it seems at first glance, however, as the y coordinate is the vertical separation between the platform and the target. If the target is submerged, this value is unknown, but the altitude of the platform is known. The target can be no closer to the platform than this altitude, since the target is physically unable to rise above the ocean surface. As an example, if a target depth of 200 feet is assumed, the initial y separation of 1000 feet forces the altitude in this simulation to be 800 feet. Using this value as a bound on y, it can be seen in Figure 4.4 that the y calculated position is somewhat improved. Accordingly, it seems that the particular arrangements of sensors shown in Figure 1.1 does show promise in the problem of utilizing this geometry of Multiple MAD for accurate localization purposes. Accordingly, variations of parameters were tried in order to determine what effect changing them would have. In all cases, parameters not specifically mentioned in a section will retain the values as for the initial run described above. The singularity-like behavior occuring at various places on the graphs are a result of the determinant of the denominator in the Cramer's Rule subroutine having a value close to zero.

B. HEADING CHANGES

In this section, the localization process will be investigated as a function of heading. Compared to the initial conditions described above, the crossover points were found to be moved further out past 5000 feet prior to CPA on a heading of 0°. While a specific example of this is not shown, the behavior of the crossover points for the platform headings of 30° and 60° may be observed by comparing Figures 4.1 - 4.4 to the respective portions of Figure 4.5 . The x position estimates were better for a North-South heading than for any other platform heading, although the estimates on East-West headings were also very good. For a 90° heading there are two crossover points located equidistant at 1200 feet from CPA. Calculated positions on all headings are generally good, excepting those around 60°, where the x estimates shows a large discrepancy at distances greater than 3000 feet prior to CPA. The y estimates are not significantly affected, but the z estimates are good only within ± 1200 feet of CPA. Nevertheless, even in this worst-case situation, reasonable calculated coordinates are available for localization along a significant portion of the track.

As shown in Figures 4.5 - 4.8, headings in the quadrant 0° - 90° reasonably describe what happens on any heading, as the graphs for analogous headings (60°, 120°, 240°, and 300° in this example) have similar shapes, although the orientation may differ between quadrants. These similarities are apparently due to the fact that analogous heading in quadrants 1 and 3 and quadrants 2 and 4 are in fact reciprocal headings, and so one would expect the graphs to maintain the same shape; however, the ending point of one would represent the starting point of the other. The relationship between adjacent quadrants probably exists for similar reasons.

C. TARGET DIPOLE MOMENT CHANGES

In this section, the horizontal component of $\vec{m}$ was oriented in the 0° (magnetic North) direction vice 355°, and the vertical component of $\vec{m}$ was varied. The best positioning was obtained for $\vec{m}$ oriented vertically up,i.e., no horizontal component, and an example for that case is shown in figure 4.9 . Poor positioning was obtained for the vertical component of $\vec{m}$ pointing between the horizontal and 45° up. The worst case shown is Figure 4.10, where $\vec{m}$ is oriented horizontally pointing due north. With the moment oriented at an angle below the horizontal, reasonable positioning estimates were obtained. There was less variability in positioning for moments pointing below the horizontal than for those pointing above the horizontal.

The results for variations in the orientation of the target dipole moment suggests that the positioning estimates are less sensitive to moment orientation changes in the horizontal plane as compared to the vertical plane. For example, by keeping the vertical component of $\vec{m}$ constant at an angle of 50° down while varying the horizontal component direction it was found that the horizontal component affected the positioning much less than variations in the vertical. Illustrated in Figure 4.11, the x estimates of position improve as the horizontal component approaches the platform heading, while the y estimates improve as the horizontal component approaches alignment with magnetic North. The latter case is not shown.

D. CHANGES IN SENSOR SPACING

Changing the inter-sensor spacing had at most a weak effect upon the position output, even with spacing as short as 5 feet or as long as 200 feet. This would be expected in a computer simulation where sufficient precision is

available. For a spacing as great as 1000 feet some effect became noticable. In the real world, sensor spacing would influence the localization estimates.

E. ALTITUDE CHANGES

For a straight overhead pass, positioning information appears best if a pass with a 500 to 1000 foot separation between the target and the platform is made at CPA. At greater altitudes, the accuracy suffers slightly with the increase in altitude, while at lesser ranges, usable positioning information is obtained only relatively close to CPA. For example, for a vertical separation of 400 feet, positioning data was very inaccurate at slant ranges greater than 1000 feet.

F. NON-OVERHEAD PASSES

This program gives poor results for passes that are not directly over or directly off to one side of the target. The worst situation observed in any variation is illustrated in Figure 4.12 . To generate this figure, a pass was made such that the actual x coordinate was approximately equal to the actual y coordinate. The best positioning was achieved by overhead passes, but low off to the side passes can yield good results as well.

G. AVERAGES

The averages resulting from a number of runs are presented in Table I . These selected averages, covering all of the variations discussed in this chapter, show little consistency. This is probably due in part to the existence of the singularities. The variability is also a function of the limits of travel for the runs. Nevertheless, for most

TABLE I

Values of x and y Averaged over a Run

run	hdg	d	$\vec{m}$ vert	$\vec{m}$ horiz	avg x act	avg x calc	avg y act	avg y calc
1	30°	50	50°	-5°	0	42	1000	480
2	0°	50	50°	-5°	0	-17	1000	651
3	15°	50	50°	-5°	0	19	1000	548
4	45°	50	50°	-5°	0	-296	1000	893
5	60°	50	50°	-5°	0	302	1000	-248
6	75°	50	50°	-5°	0	244	1000	-185
7	90°	50	50°	-5°	0	222	1000	-169
8	120°	50	50°	-5°	0	326	1000	-217
9	150°	50	50°	-5°	0	96	1000	385
10	180°	50	50°	-5°	0	11	1000	539
11	225°	50	50°	-5°	0	244	1000	825
12	270°	50	50°	-5°	0	-219	1000	-167
13	315°	50	50°	-5°	0	-10	1000	585
14	30°	50	50°	0°	0	26	1000	494
15	30°	50	70°	0°	0	-96	1000	1025
16	90°	50	70°	0°	0	384	1000	-532
17	30°	50	-90°	0°	0	199	1000	-796
18	30°	50	-45°	0°	0	-206	1000	-674
19	30°	50	0°	0°	0	-8	1000	-18
20	30°	50	90°	0°	0	199	1000	-797
21	30°	50	50°	5°	0	79	1000	462
22	30°	50	50°	30°	0	102	1000	631
23	30°	50	50°	90°	0	66	1000	993
24	30°	50	50°	-45°	0	383	1000	-270
25	30°	50	50°	-90°	0	376	1000	-321
26	30°	50	50°	-5°	436	412	900	1690
27	30°	50	50°	-5°	600	-180	800	199
28	30°	50	50°	-5°	714	-116	700	848
29	30°	50	50°	-5°	800	-255	600	911
30	30°	50	50°	-5°	867	-501	500	440
31	30°	50	50°	-5°	917	-412	400	-423
32	30°	50	50°	-5°	954	1147	300	-2025
33	30°	50	50°	-5°	980	944	200	-1861
34	30°	50	50°	-5°	994	797	100	-1665
35	30°	50	50°	-5°	1000	813	0	-1471
36	30°	50	50°	-5°	-800	-478	600	538
37	30°	50	50°	-5°	-600	-345	800	492
38	30°	5	50°	-5°	0	37	1000	466
39	30°	10	50°	-5°	0	37	1000	461
40	30°	25	50°	-5°	0	38	1000	470
41	30°	75	50°	-5°	0	45	1000	490
42	30°	100	50°	-5°	0	48	1000	500
43	30°	1000	50°	-5°	0	193	1000	759
44	30°	50	50°	-5°	0	-66	800	662
45	30°	50	50°	-5°	0	-238	600	1058
46	30°	50	50°	-5°	0	-870	400	1757
47	30°	50	50°	-5°	0	390	200	263
48	30°	50	50°	-5°	0	108	1200	372
49	30°	50	50°	-5°	0	219	1400	246
50	30°	50	50°	-5°	0	266	1600	192
51	30°	50	50°	-5°	0	313	1800	152
52	30°	50	50°	-5°	0	410	2000	72
53	30°	50	-45°	45°	0	275	1000	-344
54	30°	50	0°	45°	0	138	1000	140
55	30°	50	50°	45°	0	-226	1000	693
56	30°	50	-45°	-45°	0	-584	1000	-785
57	30°	50	0°	-45°	0	-802	1000	19
58	30°	50	45°	-45°	0	332	1000	-201

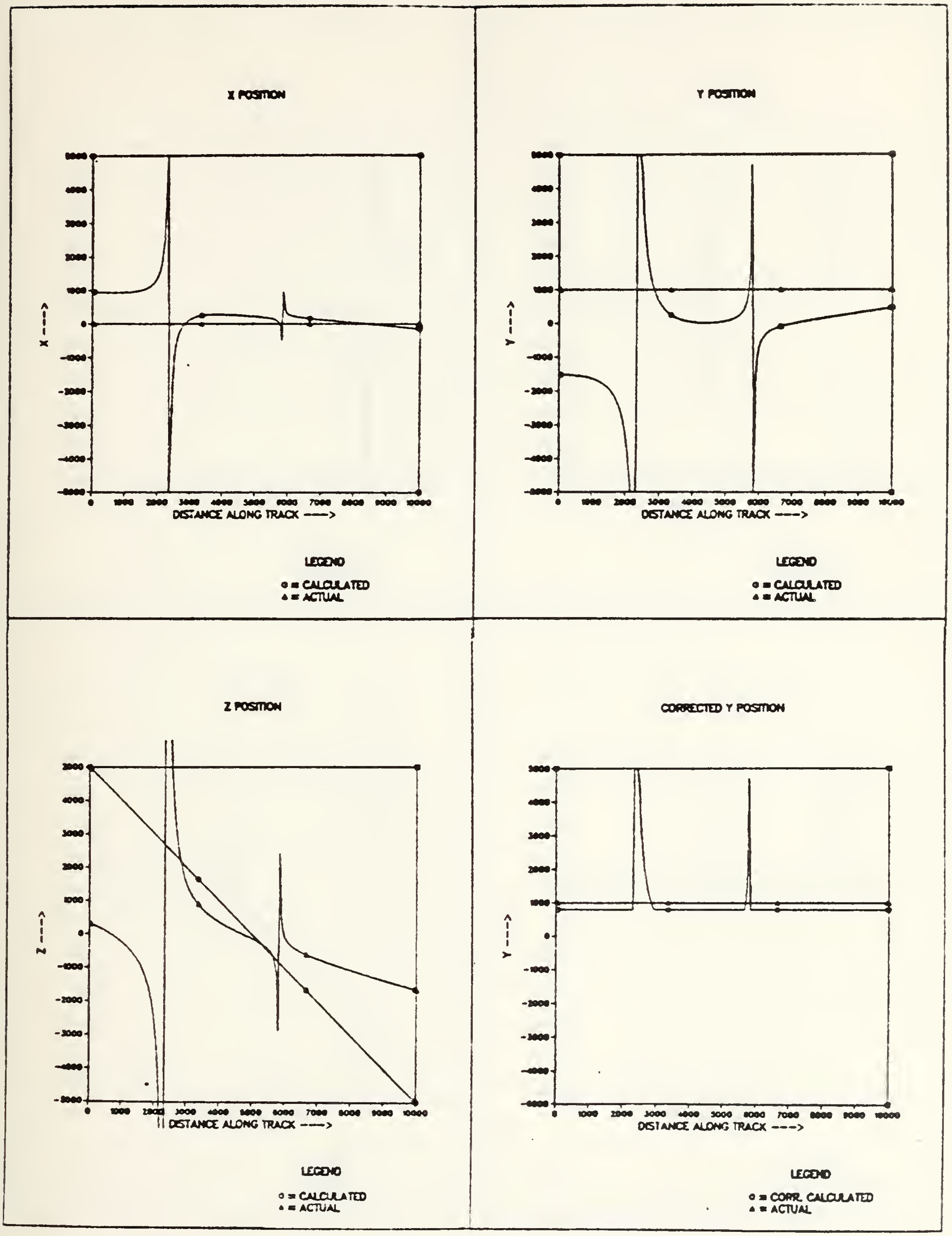
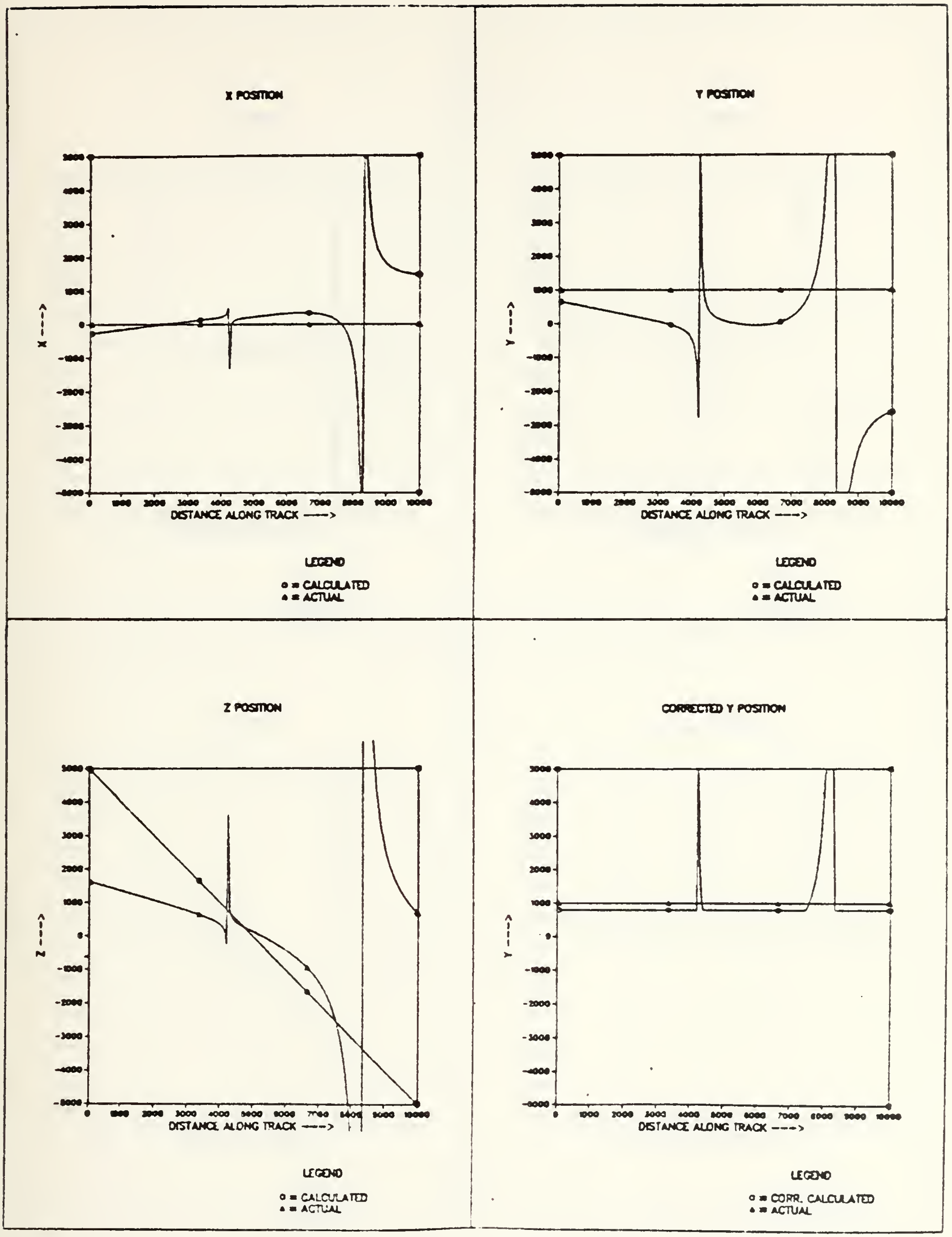


Figure 4.5 Platform Heading 60° Magnetic.



X POSITION

Y POSITION

Z POSITION

CORRECTED Y POSITION

Figure 4.6 Platform Heading 120° Magnetic.

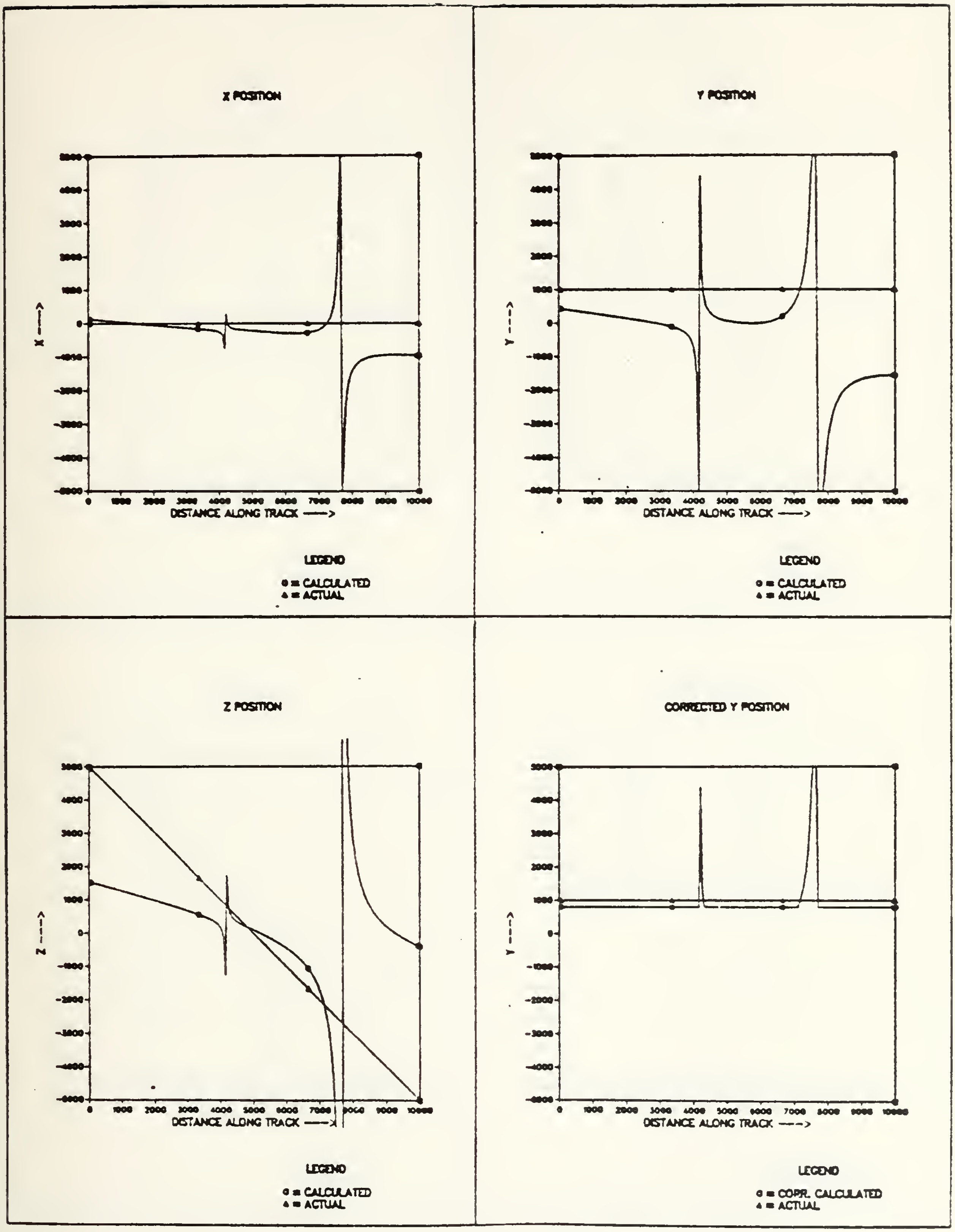


Figure 4.7 Platform Heading 240° Magnetic.

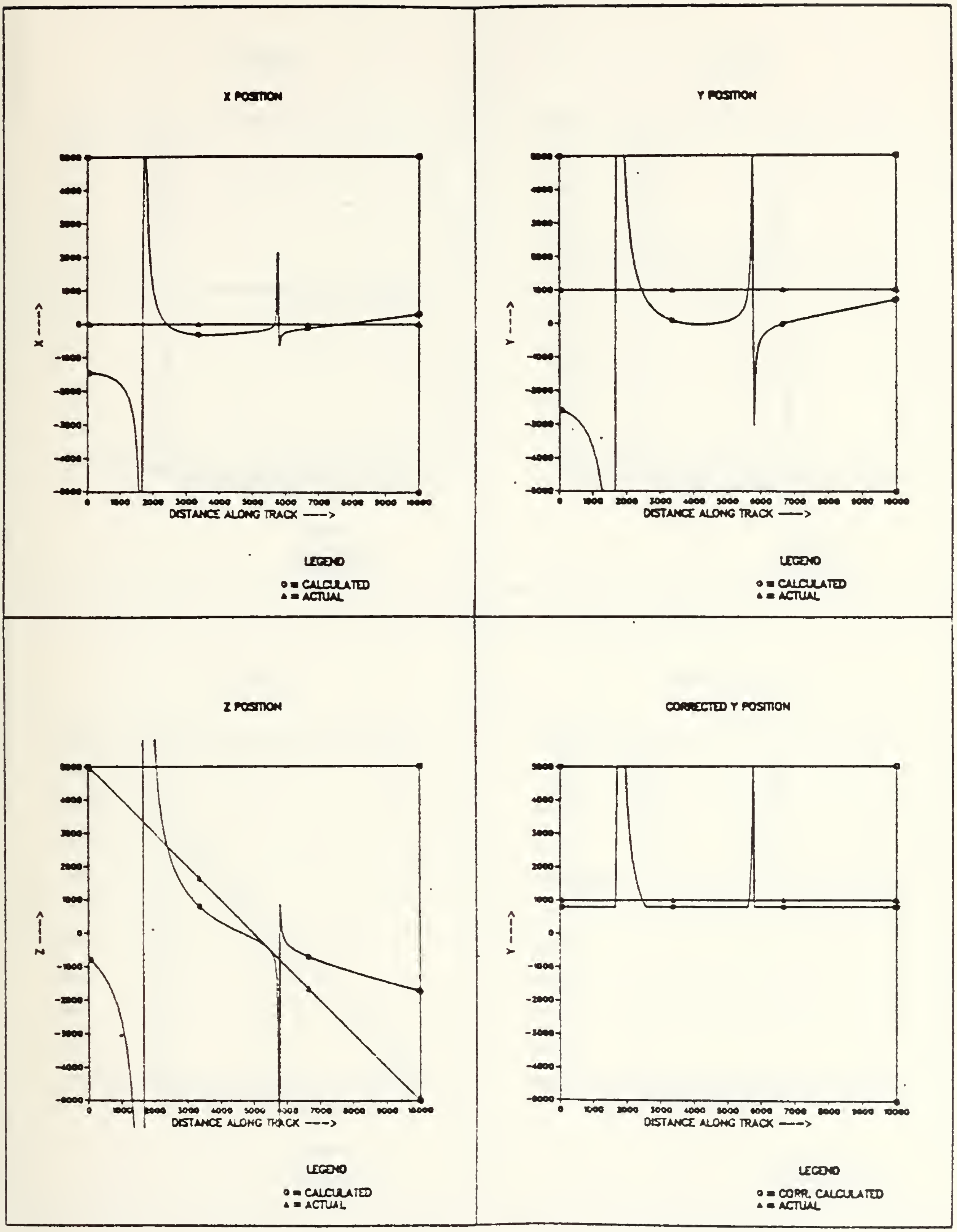


Figure 4.8 Platform Heading 300° Magnetic.

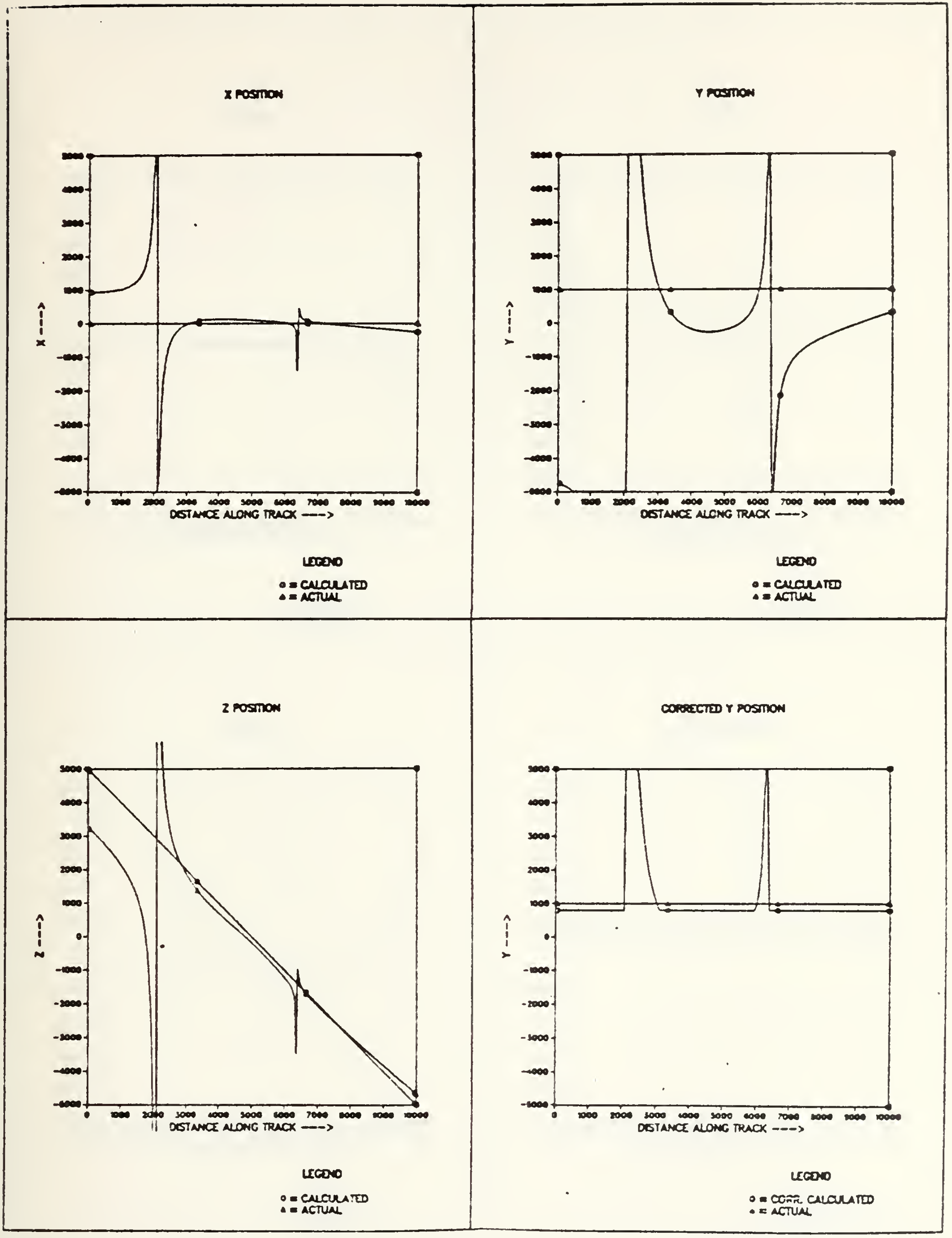


Figure 4.9 $\vec{m}$ Vertically Up.

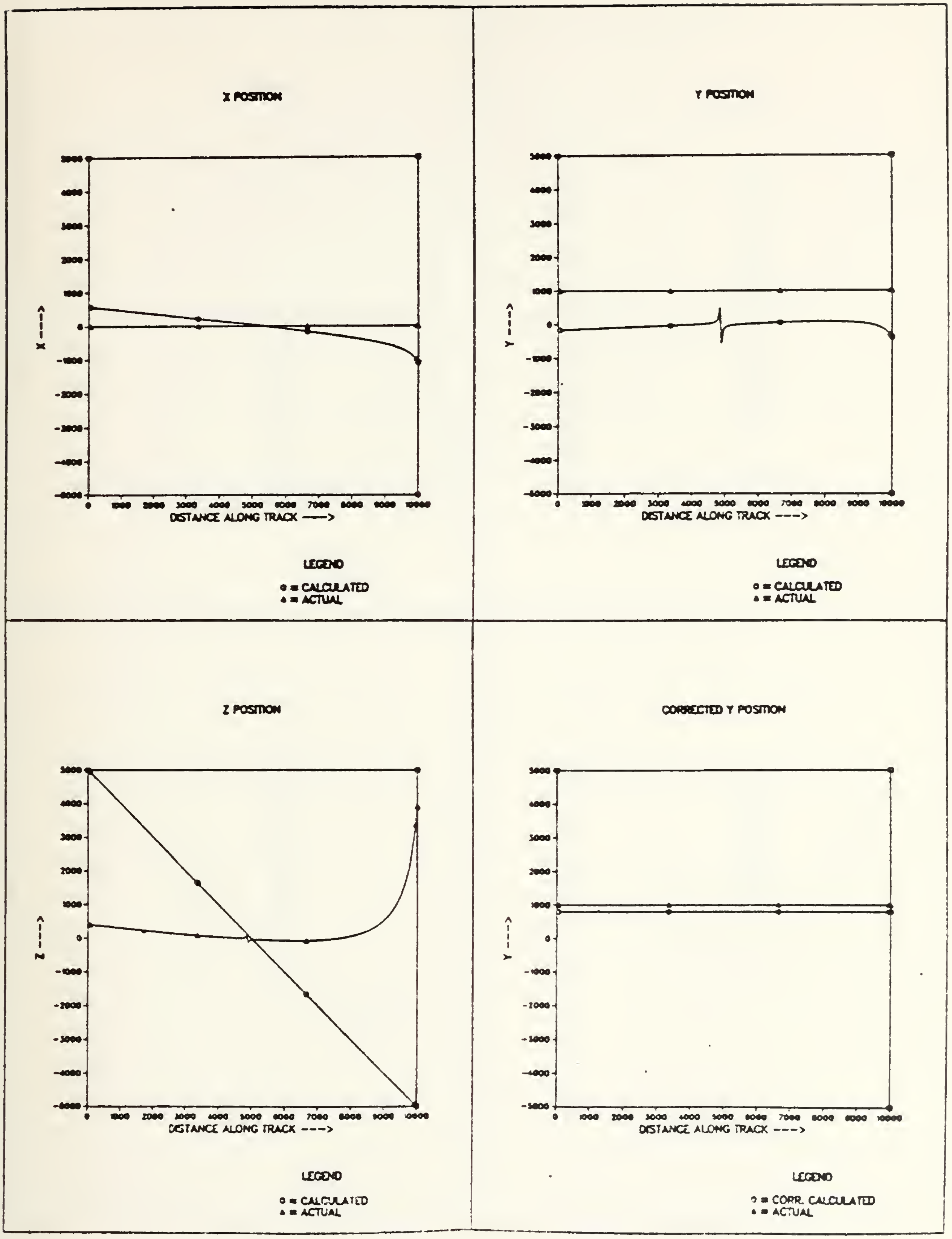


Figure 4.10 $\vec{m}$ Horizontal and Oriented to Magnetic North.

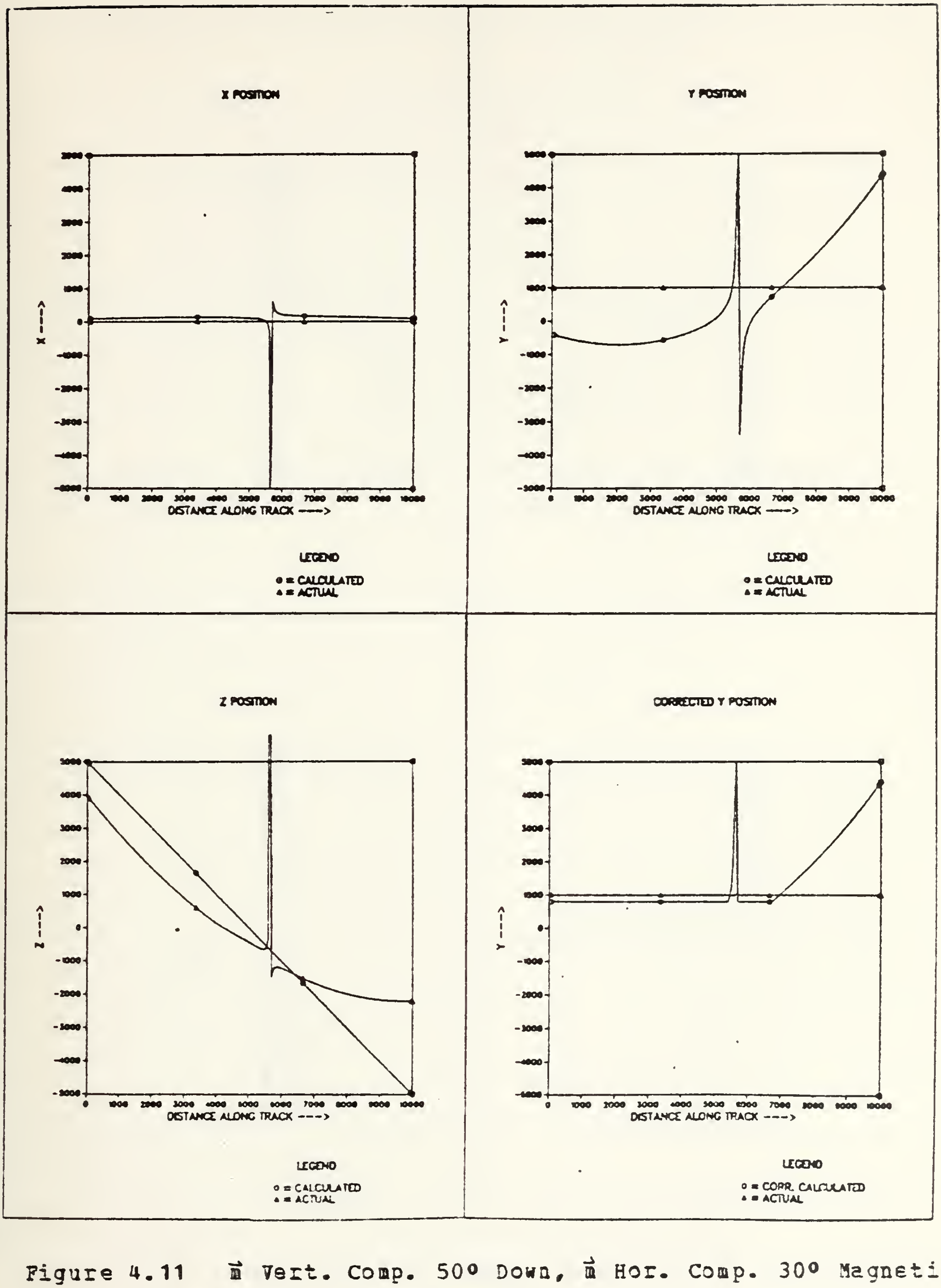


Figure 4.11 $\vec{m}$ Vert. Comp. 50° Down, $\vec{m}$ Hor. Comp. 30° Magnetic.

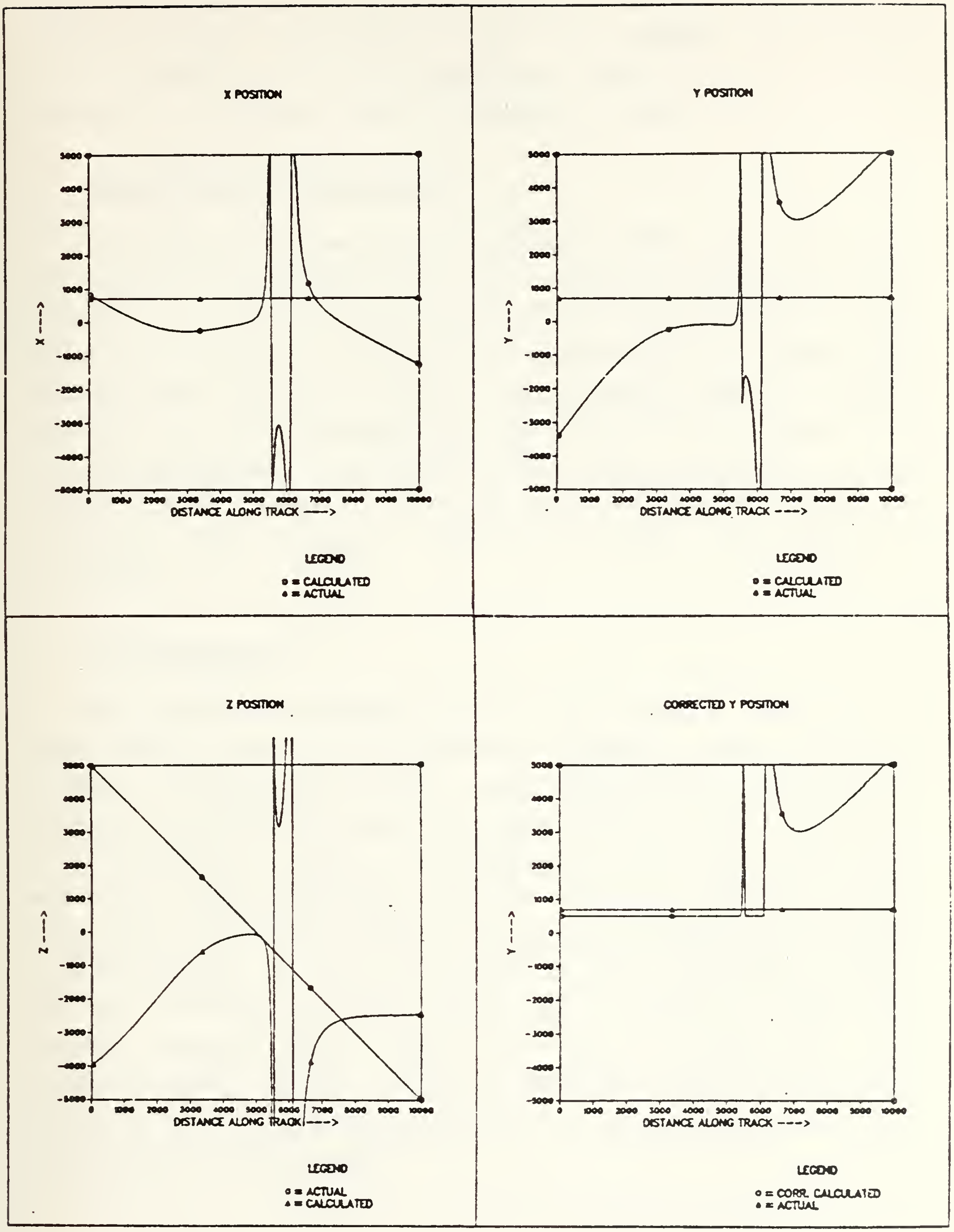


Figure 4.12 Target at 45° depres. angle rel. to platform at CPA.

cases, they do show reasonable values, and by themselves do provide limited localization data. The problem for inflight analysis would be to determine the bounds of a run in the absence of any other target positioning data.

H. IMPORTANCE OF CONSISTENCY

An important observation resulting from the analysis of the graphs, including those not shown as examples in this chapter, is that the rate of change of the calculated coordinates gives some indication of the validity of the positioning data. If the x and y graphs have a slope close to zero, and if the z graph has a slope of -45^o for the calculated coordinate, then positioning data is likely to be comparatively accurate. If all three graphs fulfill this condition simultaneously, then positioning data appears to be usable.

I. IMPLEMENTATION

The possible implementation of the process described in this thesis would be to assume a magnetic moment for the target, then run the signal data through the processing to generate positioning data, and then compare these results to those obtained by using the assumed moment to generate a signal via this simulation. If the agreement between results were good, one could assume that one had assumed the correct target moment; if not, one would assume a different moment, and go through the process again, until one had a reasonable target moment, which would be valid at least until the target changed course. Using a computer these comparisons could be made reasonably quickly, so this process might have operational possibilities.

V. CONCLUSION

For the assumptions made, the positioning estimates are reasonable. Also, the averages taken of the three coordinates as calculated throughout the various runs show that the x calculation averages are in general at worst within 500 feet of the actual x value, while the worst y calculation averages tend to be within 1000 feet of the actual y value. The best calculated x and y values were within 100 and 200 feet, respectively, of the true values.

Trends observed were as follows:

1. It is best to fly the sensor platform directly over the target, or alternatively to have a 'distant' (between 1000 and 2000 foot) CPA off to one side while minimizing the target to platform separation in the y direction. The worst situation is when the target is at a point that, at CPA, has a depression angle of 45° with respect to the platform.
2. Best vertical separations are those that will yield a 500 to 1000 foot vertical distance between target and platform. Short vertical separations should be avoided if good localization is desired.
3. Best results are obtained if headings are northerly or southerly. Poor results are achieved with headings close to 60°, 120°, 240°, and 300°. East-West headings, while not as good as North-South, do provide good positioning data close to CPA.
4. Good results may be expected if there is a large vertical moment, especially if it is vertically up. Poor positioning results are obtained with this simulation if the moment has a large horizontal component.

5. Results are slightly better if the target moment is lined up with either the axis of flight or with the earth's field, although this effect is often dominated by the more gross effects caused by the other variables.

A suggested approach for further study is to inject noise into the simulation in order to see how robust it is in the accuracy of the position calculations. If the results remain reasonable, this approximation or an altered form of it may one day have real-world applicability.

APPENDIX A

COMPUTER PROGRAM FOR PLOT OUTPUT

```
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCM1000010
C                                                                      CM1000020
C   THIS PROGRAM PRODUCES A GRAPHICAL OUTPUT OF ACTUAL CARTESIAN       CM1000030
C   POSITIONS AND CALCULATED CARTESIAN POSITIONS AS THE PLATFORM       CM1000040
C   MAKES A STRAIGHT AND LEVEL PASS PAST THE TARGET. CPA OCCURS        CM1000050
C   AT 5000 FEET ALONG TRACK.                                          CM1000060
C                                                                      CM1000070
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCM1000080
      REAL*8 MX,MY,MZ,EX,EY,EZ,X,Y,Z,D,S,RG,S2S3,S2S5,G,PI,RADN,RTHETA, M1000090
     *RPHEE,S3S1,S5S1,S3S4,A1,A2,A3,C1,C2,C3,C4,C5,C6,C7,C8,C9,K1,K2,K3,M1000100
     *S5S4,V1,V2,V3,V4,V5,V6,RALFA,RBETA                                M1000110
      DIMENSION X(5),Y(5),Z(5),S(5),RNG(5),ANUM(5),ZEIT(202),RX(201),RY(M1000120
     *201),RZ(201),CX(201),CY(201),CZ(201),RR(201),COA(3),COB(3),YC(201)M1000130
      N = 0                                                             M1000140
C                                                                       M1000150
C   PLATFORM HEADING, EARTH'S MAGNETIC FIELD VECTOR, AND TARGET         M1000160
C   MAGNETIC FIELD VECTOR, WITH A SET MAGNITUDE OF 5 GAMMA-FOOT         M1000170
C   CUBED, ARE INTERACTIVELY ENTERED HERE.                              M1000180
C                                                                       M1000190
      WRITE (6,50)                                                      M1000200
   50 FORMAT ('  ENTER HEADING OF PLATFORM, 0 - 360 DEGREES.')          M1000210
      READ (5,*) HEAD                                                   M1000220
      WRITE (6,51)                                                      M1000230
   51 FORMAT ('  ENTER DIP ANGLE OF EARTH MAGNETIC FIELD, + = DOWN.')   M1000240
      READ (5,*) EDIP                                                   M1000250
      WRITE (6,52)                                                      M1000260
   52 FORMAT ('  ENTER DIP ANGLE OF TARGET MAGNETIC MOMENT, + = DOWN.') M1000270
      READ (5,*) DIPM                                                   M1000280
      WRITE (6,53)                                                      M1000290
   53 FORMAT ('  ENTER HORIZONTAL DIRECTION OF TARGET MAGNETIC MOMENT'/'M1000300
     *-90 = WEST, 0 = NORTH, +90 = EAST.')                              M1000310
      READ (5,*) HORM                                                   M1000320
      THETA = EDIP                                                      M1000330
      PHEE = HEAD                                                       M1000340
      ALFA = DIPM                                                       M1000350
      BETA = HEAD - HORM                                                M1000360
      PI = 4.D0 * DATAN(1.D0)                                           M1000370
      RADN = PI / 180.D0                                                M1000380
      RTHETA = THETA * RADN                                             M1000390
      RPHEE = PHEE * RADN                                               M1000400
      RALFA = ALFA * RADN                                               M1000410
      RBETA = BETA * RADN                                               M1000420
C                                                                       M1000430
C   THE X,Y,Z COMPONENTS OF THE DIPOLE MOMENT AND THE EARTH'S FIELD     M1000440
C   UNIT VECTOR ARE CALCULATED AT THIS POINT.                           M1000450
C                                                                       M1000460
      MX = (-5.D0)*(DCOS(RALFA))*(DSIN(RBETA))                          M1000470
      MY = 5.D0*(DSIN(RALFA))                                           M1000480
```

```
      MZ =  5.DO*(DCOS(RALFA))*(DCOS(RBETA))                             M1000490
      EX =  -(DCOS(RTHETA))*(DSIN(RPHEE))                                M1000500
      EY =  DSIN(RTHETA)                                                 M1000510
      EZ =  (DCOS(RTHETA))*(DCOS(RPHEE))                                 M1000520
      G  = (MX*EX)+(MY*EY)+(MZ*EZ)                                       M1000530
      V1 =  (3.DO*MX*EX) - G                                             M1000540
      V2 =  3.DO  *  ((MX*EY)+(MY*EX))                                   M1000550
      V3 =  (3.DO*MY*EY) - G                                             M1000560
      V4 =  3.DO  *  ((MY*EZ)+(MZ*EY))                                   M1000570
      V5 =  (3.DO*MZ*EZ) - G                                             M1000580
      V6 =  3.DO  *  ((MX*EZ)+(MZ*EX))                                   M1000590
C                                                                        M1000600
C    THE INITIAL POSITION OF SENSOR 1, AS WELL AS THE SENSOR SPACING     M1000610
C    DISTANCE "D" IS SET HERE - IF A CHANGE IS DESIRED, THE PROGRAM      M1000620
C    MUST BE ALTERED AT THIS POINT.                                      M1000630
C                                                                        M1000640
      X(1)  =      0.DO                                                  M1000650
      Y(1)  = -1000.DO                                                   M1000660
      Z(1)  = -5000.DO                                                   M1000670
      D = 50.DO                                                          M1000680
      COB(1)  =  0.0                                                     M1000690
      COA(1)  =  5000.0                                                  M1000700
      COB(2)  =  10000.0                                                 M1000710
      COA(2)  =  5000.0                                                  M1000720
      COB(3)  =  10000.0                                                 M1000730
      COA(3)  =  -5000.0                                                 M1000740
C                                                                        M1000750
C    THE INITIAL POSITIONS OF SENSORS 2 - 5 ARE CALCULATED HERE.         M1000760
C                                                                        M1000770
      X(2)  =  X(1)                                                      M1000780
      Y(2)  =  Y(1) - D                                                  M1000790
      X(3)  =  X(1) - D                                                  M1000800
      Y(3)  =  Y(1)                                                      M1000810
      X(4)  =  X(1)                                                      M1000820
      Y(4)  =  Y(1)                                                      M1000830
      X(5)  =  X(1) + D                                                  M1000840
      Y(5)  =  Y(1)                                                      M1000850
   30 Z(2)  =  Z(1)                                                      M1000860
      Z(3)  =  Z(1)                                                      M1000870
      Z(4)  =  Z(1) - D                                                  M1000880
      Z(5)  =  Z(1)                                                      M1000890
C                                                                        M1000900
C    THE ACTUAL SIGNAL, BASED ON THE DIPOLE EQUATION, TO BE RECEIVED     M1000910
C    AT EACH OF THE FIVE SENSORS IS CALCULATED AT THIS POINT.            M1000920
C                                                                        M1000930
      DO 20  J=1,5                                                       M1000940
      RG =  DSQRT((X(J)**2) + (Y(J)**2) + (Z(J)**2))                     M1000950
      DOTME  =  (MX*EX)+(MY*EY)+(MZ*EZ)                                  M1000960
```

```
      DOTMR  = (MX*X(J))+(MY*Y(J))+(MZ*Z(J))
      DOTRE  = (X(J)*EX)+(Y(J)*EY)+(Z(J)*EZ)
      TERM1  = (-DOTME)*(RG**2)
      TERM2  = 3.0*DOTMR*DOTRE
      ANUM(J) =  TERM1 + TERM2
      TERM3  = 1.D8 / (RG**5)
      S(J)  = TERM3 * (ANUM(J))
      RNG(J) = RG
   20 CONTINUE
C
C    DIFFERENCES OF RECEIVED SIGNAL ARE TAKEN, IN ORDER TO ELIMINATE
C    SQUARE AND HIGHER ORDER TERMS. THE DENOMINATOR IS ASSUMED TO BE
C    THE SAME FOR THIS APPROXIMATION.
C
      S2S3  = S(2) - S(3)
      S2S5  = S(2) - S(5)
      S3S1  = S(3) - S(1)
      S5S1  = S(5) - S(1)
      S3S4  = S(3) - S(4)
      S5S4  = S(5) - S(4)
C
C    THE RATIOS OF THE ABOVE DIFFERENCES ARE TAKEN IN ORDER TO
C    ELIMINATE THE FIFTH-ORDER TERM IN THE DENOMINATOR.
C
      A1 = S2S3/S2S5
      A2 = S3S1/S5S1
      A3 = S3S4/S5S4
C
C    EQUATION 3.1 COEFFICIENTS
C
      C1 = (A1*(V2+(2.D0*V1))) - (V2-(2.D0*V1))
      C2 = (A1*((2.D0*V3)+V2)) - ((2.D0*V3)-V2)
      C3 = (A1*(V4+V6)) - (V4-V6)
      K1 = ((V1-V3) - (A1*(V1-V3))) * D
C
C    EQUATION 3.2 COEFFICIENTS
C
      C4 = (A2*2.D0*V1) + (2.D0*V1)
      C5 = (A2*V2) + V2
      C6 = (A2*V6) + V6
      K2 = (V1 - (A2*V1)) * D
C
C    EQUATION 3.2 COEFFICIENTS
C
      C7 = (A3*(V6+(2.D0*V1))) - (V6-(2.D0*V1))
      C8 = (A3*(V4+V2)) - (V4-V2)
      C9 = (A3*((2.D0*V5)+V6)) - ((2.D0*V5)-V6)
      K3 = ((V1-V5) - (A3*(V1-V5))) * D
```

```
C
C    THE COEFFICIENTS ARE SENT TO THE CRAMER'S RULE SUBROUTINE.
C
      CALL SUBMAT(C1,C2,C3,K1,C4,C5,C6,K2,C7,C8,C9,K3,XX,YY,ZZ)
      IF (XX.LT.5000.0) GO TO 80
      XX = 5000.0
      GO TO 81
   80 IF (XX.GT.-5000.0) GO TO 81
      XX = -5000.0
   81 RX(N) = X(1)
      CX(N) = XX
      IF (YY.LT.5000.0) GO TO 82
      YY = 5000.0
      GO TO 83
   82 IF (YY.GT.-5000.0) GO TO 83
      YY = -5000.0
   83 RY(N) = Y(1)
      CY(N) = YY
      RZ(N) = Z(1) + 5000.0
      RR(N) = Z(1)
      CZ(N) = ZZ
      N = N + 1
      ZEIT(N) = N
      IF (Z(1).GE.5000.D0) GO TO 40
C
C    THE PLATFORM IS ADVANCED ONE TIME STEP IN POSITION. IF A DIFFERENT
C    SIZE STEP IS DESIRED, IT MUST BE ADJUSTED HERE.
C
      Z(1) = Z(1) +  50.D0
      GO TO 30
C
C    THIS IS THE PLOTTING ROUTINE.
C
   40 DO 97 K=1,200
      RX(K) = -RX(K)
      RY(K) = -RY(K)
      RR(K) = -RR(K)
      CX(K) = -CX(K)
      CY(K) = -CY(K)
      CZ(K) = -CZ(K)
      HOCH = -Y(1) - 200.0
      IF (CY(K).LT.HOCH) GO TO 96
      YC(K) = CY(K)
      GO TO 97
   96 YC(K) = HOCH
   97 CONTINUE
      CALL TEK618
      CALL PLOTD(COB,COA,3,.FALSE.,'LINLIN',' $','X POSITION$','DISTANCE
```

```
     * ALONG TRACK --->$','X --->$')                                    M1001930
      CALL PLOTD(RZ,CX,200,.FALSE.,'LINLIN','CALCULATED$','X POSITION$',M1001940
     *'DISTANCE ALONG TRACK --->$','X --->$')                           M1001950
      CALL PLOTD(RZ,RX,200,.TRUE.,'LINLIN','ACTUAL$','X POSITION$','DISTM1001960
     *ANCE ALONG TRACK --->$','X --->$')                                M1001970
      CALL PLOTD(COB,COA,3,.FALSE.,'LINLIN',' $','Y POSITION$','DISTANCEM1001980
     * ALONG TRACK --->$','Y --->$')                                    M1001990
      CALL PLOTD(RZ,CY,200,.FALSE.,'LINLIN','CALCULATED$','Y POSITION$',M1002000
     *'DISTANCE ALONG TRACK --->$','Y --->$')                           M1002010
      CALL PLOTD(RZ,RY,200,.TRUE.,'LINLIN','ACTUAL$','Y POSITION$','DISTM1002020
     *ANCE ALONG TRACK --->$','Y --->$')                                M1002030
      CALL PLOTD(COB,COA,3,.FALSE.,'LINLIN',' $','Z POSITION$','DISTANCEM1002040
     * ALONG TRACK --->$','Z --->$')                                    M1002050
      CALL PLOTD(RZ,CZ,200,.FALSE.,'LINLIN','CALCULATED$','Z POSITION$',M1002060
     *'DISTANCE ALONG TRACK --->$','Z --->$')                           M1002070
      CALL PLOTD(RZ,RR,200,.TRUE.,'LINLIN','ACTUAL$','Z POSITION$','DISTM1002080
     *ANCE ALONG TRACK --->$','Z --->$')                                M1002090
      CALL PLOTD(COB,COA,3,.FALSE.,'LINLIN',' $','CORRECTED Y POSITION$'M1002100
     *,'DISTANCE ALONG TRACK --->$','Y --->$')                          M1002110
      CALL PLOTD(RZ,YC,200,.FALSE.,'LINLIN','CORR. CALCULATED$','CORR. YM1002120
     * POSITION$','DISTANCE ALONG TRACK --->$','Y --->$')               M1002130
      CALL PLOTD(RZ,RY,200,.TRUE.,'LINLIN','ACTUAL$','CORR. Y POSITION$'M1002140
     *,'DISTANCE ALONG TRACK --->$','Y --->$')                          M1002150
      CALL DONEPL                                                       M1002160
      XAV = 0.0                                                         M1002170
      YAV = 0.0                                                         M1002180
      ZAV = 0.0                                                         M1002190
      DO 45 K = 1,200                                                   M1002200
      XAV = XAV + CX(K)                                                 M1002210
      YAV = YAV + CY(K)                                                 M1002220
      ZAV = ZAV + CZ(K)                                                 M1002230
   45 CONTINUE                                                          M1002240
      AVGX = XAV/200                                                    M1002250
      AVGY = YAV/200                                                    M1002260
      AVGZ = ZAV/200                                                    M1002270
      WRITE (6,60)                                                      M1002280
      TX = -X(1)                                                        M1002290
      TY = -Y(1)                                                        M1002300
      WRITE (6,85) TX,AVGX                                              M1002310
   85 FORMAT (1X,F7.1,' IS THE TRUE X DISTANCE.'/1X,F7.1,' IS THE AVERAGM1002320
     *E CALCULATED X DISTANCE.')                                        M1002330
      WRITE (6,86) TY,AVGY                                              M1002340
   86 FORMAT (1X,F7.1,' IS THE TRUE Y DISTANCE,'/1X,F7.1,' IS THE AVERAGM1002350
     *E CALCULATED Y DISTANCE.')                                        M1002360
      WRITE (6,73) AVGZ                                                 M1002370
   73 FORMAT (1X,F7.1,' IS THE AVERAGE CALCULATED Z VALUE.')            M1002380
      WRITE (6,87) HEAD                                                 M1002390
   87 FORMAT (1X,F7.1,' IS THE PLATFORM HEADING.')                      M1002400
```

```
      WRITE (6,88) EDIP                                                 M1002410
   88 FORMAT (1X,F7.1,' IS THE DIP ANGLE OF THE EARTH',1H','S MAGNETIC FM1002420
     *IELD, + IS DOWN.')                                                M1002430
      DEG = 360.C + HORM                                                M1002440
      IF (DEG.LT.360.0) GO TO 84                                        M1002450
      DEG = DEG - 360.0                                                 M1002460
   84 WRITE (6,89) DIPM,DEG                                             M1002470
   89 FORMAT (1X,F7.1,' IS THE DIP ANGLE OF THE TARGET',1H','S DIPCLE MOM1002480
     *MENT, + IS DOWN.'/1X,F7.1,' IS THE HORIZONTAL COMPONENT OF THE SAMM1002490
     *E.')                                                              M1002500
      WRITE (6,60)                                                      M1002510
   60 FORMAT (' ')                                                      M1002520
      STOP                                                              M1002530
      END                                                               M1002540
C                                                                       M1002550
C    CRAMER'S RULE SUBROUTINE                                           M1002560
C                                                                       M1002570
      SUBROUTINE SUBMAT(AX,AY,AZ,AA,BX,BY,BZ,BB,CX,CY,CZ,CC,XX,YY,ZZ)   M1002580
      REAL*8 AX,AY,AZ,AA,BX,BY,BZ,BB,CX,CY,CZ,CC,XX,YY,ZZ,DD            M1002590
    7 FORMAT ('+',26X,'MATRIX CANNOT BE CALCULATED.')                   M1002600
      DD = AX*((BY*CZ)-(BZ*CY)) - BX*((AY*CZ)-(AZ*CY)) + CX*((AY*BZ)-(AZM1002610
     **BY))                                                             M1002620
      IF (DD.NE.0.D0) GO TO 98                                          M1002630
      WRITE (6,7)                                                       M1002640
      GO TO 99                                                          M1002650
   98 XX = (AA*((BY*CZ)-(BZ*CY)) - BB*((AY*CZ)-(AZ*CY)) + CC*((AY*BZ)-(AM1002660
     *Z*BY)))/DD                                                        M1002670
      YY = (AX*((BB*CZ)-(BZ*CC)) - BX*((AA*CZ)-(AZ*CC)) + CX*((AA*BZ)-(AM1002680
     *Z*BB)))/DD                                                        M1002690
      ZZ = (AX*((BY*CC)-(BB*CY)) - BX*((AY*CC)-(AA*CY)) + CX*((AY*BB)-(AM1002700
     *A*BY)))/DD                                                        M1002710
   95 RETURN                                                            M1002720
      END                                                               M1002730
```

APPENDIX B

COMPUTER PROGRAM FOR NUMERICAL OUTPUT

```
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCM1100010
C                                                                       M1100020
C   THIS PROGRAM DOES THE SAME JOB AS THE ONE IN APPENDIX A, BUT THE    M1100030
C   OUTPUT IS IN NUMERICAL FORM. IN ADDITION TO THE CCMPARATIVE         M1100040
C   PCSITIONS, TFE ACTUAL COEFFICIENTS OF THE EQUATIONS GOING INTO      M1100050
C   THE CRAMER'S RULE SUBROUTINE ARE LISTED, AS ARE RATIOS OF THEM TO   M1100060
C   ESTABLISH THAT THEY ARE INDEED LINEARLY INDEPENDENT. THE RECEIVED   M1100070
C   SIGNAL STRENGTH AT EACH SENSOR IS ALSO COMPARED TO THE EXPECTED     M1100080
C   SIGNAL STRENGTH USING THE RANGE TO SENSOR 1 AS THE COMMON           M1100090
C   DENOMINATOR FOR THE APPROXIMATION. NOTE THAT THE POSITIONS GIVEN    M1100100
C   AS THE SOLUTION HAVE THE TARGET, NOT THE PLATFORM, AS THE ORIGIN,   M1100110
C   AND THAT SAID POSITIONS ARE THE PLATFORM'S PCSITICN.                M1100120
C                                                                       M1100130
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCM1100140
      REAL*8 MX,MY,MZ,EX,EY,EZ,X,Y,Z,D,S,RG,S2S3,S2S5,G,PI,RADN,RTHETA, M1100150
     *RPHEE,S3S1,S5S1,S3S4,A1,A2,A3,C1,C2,C3,C4,C5,C6,C7,C8,C9,K1,K2,K3,M1100160
     *S5S4,V1,V2,V3,V4,V5,V6,RALFA,RBETA                                M1100170
      REAL K1K2,K1K3,K2K3                                               M1100180
      DIMENSION X(5),Y(5),Z(5),S(5),RNG(5),ANUM(5)                      M1100190
C                                                                       M1100200
C   TFE SAME PARAMETERS AS FOR THE PROGRAM IN APPENDIX A ARE ENTERED    M1100210
C   FERE. THE ONLY DIFFERENCE IS THAT BETA, THE DIPOLE HORIZONTAL       M1100220
C   CCMPONENT, IS INSERTED IN DEGREES COUNTER-CLOCKWISE FROM THE        M1100230
C   PLATFORM HEACING. ALL PARAMETER CHANGES IN THIS PROGRAM ARE NON-    M1100240
C   INTERACTIVE, AND MUST BE CHANGED IN THE PROGRAM ITSELF.             M1100250
C                                                                       M1100260
      PFEE = 30.                                                        M1100270
      THETA = 70.                                                       M1100280
      ALFA = 50.                                                        M1100290
      BETA = 35.                                                        M1100300
      PI = 4.D0 * DATAN(1.D0)                                           M1100310
      RADN = PI / 180.D0                                                M1100320
      RTHETA = TFETA * RADN                                             M1100330
      RPHEE = PHEE * RADN                                               M1100340
      RALFA = ALFA * RADN                                               M1100350
      RBETA = BETA * RADN                                               M1100360
      MX = (-5.D0)*(DCOS(RALFA))*(DSIN(RBETA))                          M1100370
      MY = 5.D0*(DSIN(RALFA))                                           M1100380
      MZ = 5.D0*(DCOS(RALFA))*(DCOS(RBETA))                             M1100390
      EX = -(DCOS(RTHETA))*(DSIN(RPHEE))                                M1100400
      EY = DSIN(RTHETA)                                                 M1100410
      EZ = (DCOS(RTHETA))*(DCOS(RPHEE))                                 M1100420
      G = (MX*EX)+(MY*EY)+(MZ*EZ)                                       M1100430
      V1 = (3.D0*MX*EX) - G                                             M1100440
      V2 = 3.D0 * ((MX*EY)+(MY*EX))                                     M1100450
      V3 = (3.D0*MY*EY) - G                                             M1100460
      V4 = 3.D0 * ((MY*EZ)+(MZ*EY))                                     M1100470
      V5 = (3.D0*MZ*EZ) - G                                             M1100480
```

```
      V6 = 3.D0 * ((MX*EZ)+(MZ*EX))                                     M1100490
C                                                                       M1100500
C   AS IN THE OTHER PROGRAM, SENSOR 1 INITIAL POSITION ANC THE SENSOR   M1100510
C   SPACING IS SET AT THIS POINT.                                       M1100520
C                                                                       M1100530
      X(1) = 0.D0                                                       M1100540
      Y(1) = -1000.D0                                                   M1100550
      Z(1) = -50C0.D0                                                   M1100560
      D = 50.D0                                                         M1100570
      X(2) = X(1)                                                       M1100580
      Y(2) = Y(1) - D                                                   M1100590
      X(3) = X(1) - D                                                   M1100600
      Y(3) = Y(1)                                                       M1100610
      X(4) = X(1)                                                       M1100620
      Y(4) = Y(1)                                                       M1100630
      X(5) = X(1) + D                                                   M1100640
      Y(5) = Y(1)                                                       M1100650
   30 Z(2) = Z(1)                                                       M1100660
      Z(3) = Z(1)                                                       M1100670
      Z(4) = Z(1) - D                                                   M1100680
      Z(5) = Z(1)                                                       M1100690
      DO 20 J=1,5                                                       M1100700
      RG = DSQRT((X(J)**2) + (Y(J)**2) + (Z(J)**2))                     M1100710
      DOTME = (MX*EX)+(MY*EY)+(MZ*EZ)                                   M1100720
      DOTMR = (MX*X(J))+(MY*Y(J))+(MZ*Z(J))                             M1100730
      DOTRE = (X(J)*EX)+(Y(J)*EY)+(Z(J)*EZ)                             M1100740
      TERM1 = (-DOTME)*(RG**2)                                          M1100750
      TERM2 = 3.C*DOTMR*DOTRE                                           M1100760
      ANUM(J) =  TERM1 + TERM2                                          M1100770
      TERM3 = 1.C8 / (RG**5)                                            M1100780
      S(J) = TERM3 * (ANUM(J))                                          M1100790
      RNG(J) = RG                                                       M1100800
   20 CCNTINUE                                                          M1100810
      AVRNG = RNC(1)                                                    M1100820
      DNOM = 1.D8 / (AVRNG**5)                                          M1100830
      S1 = ANUM(1) * DNOM                                               M1100840
      S2 = ANUM(2) * DNOM                                               M1100850
      S3 = ANUM(3) * DNOM                                               M1100860
      S4 = ANUM(4) * DNOM                                               M1100870
      S5 = ANUM(5) * DNOM                                               M1100880
      APP23 = S2 - S3                                                   M1100890
      APP25 = S2 - S5                                                   M1100900
      APP31 = S3 - S1                                                   M1100910
      APP51 = S5 - S1                                                   M1100920
      APP34 = S3 - S4                                                   M1100930
      APP54 = S5 - S4                                                   M1100940
    1 FORMAT (10X,'ACTUAL',8X,3F10.2)                                   M1100950
   70 FORMAT (' EQNS',25X,'X',9X,'Y',9X,'Z')                            M1100960
```

```
   71 FORMAT (' USED',24X,'POS',7X,'POS',7X,'POS')                      M1100970
   60 FORMAT (' ')                                                      M1100980
      S2S3 = S(2) - S(3)                                                M1100990
      S2S5 = S(2) - S(5)                                                M1101000
      S3S1 = S(3) - S(1)                                                M1101010
      S5S1 = S(5) - S(1)                                                M1101020
      S3S4 = S(3) - S(4)                                                M1101030
      S5S4 = S(5) - S(4)                                                M1101040
      A1 = S2S3/S2S5                                                    M1101050
      A2 = S3S1/S5S1                                                    M1101060
      A3 = S3S4/S5S4                                                    M1101070
      C1 = (A1*(V2+(2.D0*V1))) - (V2-(2.D0*V1))                         M1101080
      C2 = (A1*((2.D0*V3)+V2)) - ((2.D0*V3)-V2)                         M1101090
      C3 = (A1*(V4+V6)) - (V4-V6)                                       M1101100
      K1 = ((V1-V3) - (A1*(V1-V3))) * D                                 M1101110
      C4 = (A2*2.D0*V1) + (2.D0*V1)                                     M1101120
      C5 = (A2*V2) + V2                                                 M1101130
      C6 = (A2*V6) + V6                                                 M1101140
      K2 = (V1 - (A2*V1)) * D                                           M1101150
      C7 = (A3*(V6+(2.D0*V1))) - (V6-(2.D0*V1))                         M1101160
      C8 = (A3*(V4+V2)) - (V4-V2)                                       M1101170
      C9 = (A3*((2.D0*V5)+V6)) - ((2.D0*V5)-V6)                         M1101180
      K3 = ((V1-V5) - (A3*(V1-V5))) * D                                 M1101190
C                                                                       M1101200
C    THE BELOW LISTED VARIABLES ARE THE RATIOS OF THE COEFFICIENTS      M1101210
C    USED TO SHOW LINEAR INDEPENDENCE OF THE THREE EQUATIONS.           M1101220
C                                                                       M1101230
      K1K2 = K1/K2                                                      M1101240
      K1K3 = K1/K3                                                      M1101250
      K2K3 = K2/K3                                                      M1101260
      C1C4 = C1/C4                                                      M1101270
      C2C5 = C2/C5                                                      M1101280
      C3C6 = C3/C6                                                      M1101290
      C1C7 = C1/C7                                                      M1101300
      C2C8 = C2/C8                                                      M1101310
      C3C9 = C3/C9                                                      M1101320
      C4C7 = C4/C7                                                      M1101330
      C5C8 = C5/C8                                                      M1101340
      C6C9 = C6/C9                                                      M1101350
      WRITE (6,60)                                                      M1101360
      WRITE (6,60)                                                      M1101370
      WRITE (6,60)                                                      M1101380
      WRITE (6,60)                                                      M1101390
      WRITE (6,110)                                                     M1101400
  110 FORMAT (' SENSOR',7X,'SLANT',8X,'SENSED',8X,'CALCULATED',10X,'|',1M1101410
     *1X,'EQUATICN',4X,'CONSTANT'8X,'X',11X,'Y',11X,'Z')                M1101420
      WRITE (6,111)                                                     M1101430
  111 FORMAT (' NUMBER',7X,'RANGE',8X,'SIGNAL',10X,'SIGNAL',12X,'|',12X,M1101440
```

```
     *'NUMBER',16X,'COEFFICIENT COEFFICIENT COEFFICIENT')                M1101450
      WRITE (6,101) RNG(1),S(1),S1,K1,C1,C2,C3                           M1101460
      WRITE (6,102) RNG(2),S(2),S2,K2,C4,C5,C6                           M1101470
      WRITE (6,103) RNG(3),S(3),S3,K3,C7,C8,C9                           M1101480
      WRITE (6,112) RNG(4),S(4),S4                                       M1101490
      WRITE (6,114) RNG(5),S(5),S5                                       M1101500
  101 FORMAT (4X,'1',7X,F7.1,6X,E10.4,6X,E10.4,10X,'|',15X,'1',6X,E9.2,4M1101510
     *X,E9.2,3X,E9.2,3X,E9.2)                                            M1101520
  102 FORMAT (4X,'2',7X,F7.1,6X,E10.4,6X,E10.4,10X,'|',15X,'2',6X,E9.2,4M1101530
     *X,E9.2,3X,E9.2,3X,E9.2)                                            M1101540
  103 FORMAT (4X,'3',7X,F7.1,6X,E10.4,6X,E10.4,10X,'|',15X,'3',6X,E9.2,4M1101550
     *X,E9.2,3X,E9.2,3X,E9.2)                                            M1101560
  112 FORMAT (4X,'4',7X,F7.1,6X,E10.4,6X,E10.4)                          M1101570
  114 FORMAT (4X,'5',7X,F7.1,6X,E10.4,6X,E10.4)                          M1101580
      WRITE (6,60)                                                       M1101590
      WRITE (6,104)                                                      M1101600
  104 FORMAT (11X,'EQUATION',12X,'TRUE',16X,'APPROXIMATE',13X,'RATIO')   M1101610
      WRITE (6,105) S2S3,APP23,K1K2,C1C4,C2C5,C3C6                       M1101620
  105 FORMAT (13X,'S2-S3',13X,E10.4,10X,E10.4,15X,'1/2',5X,E9.2,4X,E9.2,M1101630
     *3X,E9.2,3X,E9.2)                                                   M1101640
      WRITE (6,106) S2S5,APP25,K1K3,C1C7,C2C8,C3C9                       M1101650
  106 FORMAT (13X,'S2-S5',13X,E10.4,10X,E10.4,15X,'1/3',5X,E9.2,4X,E9.2,M1101660
     *3X,E9.2,3X,E9.2)                                                   M1101670
      WRITE (6,107) S3S1,APP31,K2K3,C4C7,C5C8,C6C9                       M1101680
  107 FORMAT (13X,'S3-S1',13X,E10.4,10X,E10.4,15X,'2/3',5X,E9.2,4X,E9.2,M1101690
     *3X,E9.2,3X,E9.2)                                                   M1101700
      WRITE (6,108) S5S1,APP51                                           M1101710
  108 FORMAT (13X,'S5-S1',13X,E10.4,10X,E10.4)                           M1101720
      WRITE (6,109) S3S4,APP34                                           M1101730
  109 FORMAT (13X,'S3-S4',13X,E10.4,10X,E10.4)                           M1101740
      WRITE (6,113) S5S4,APP54                                           M1101750
  113 FORMAT (13X,'S5-S4',13X,E10.4,10X,E10.4)                           M1101760
      WRITE (6,60)                                                       M1101770
C     WRITE (6,60)                                                       M1101780
      WRITE (6,70)                                                       M1101790
      WRITE (6,71)                                                       M1101800
C     WRITE (6,60)                                                       M1101810
      WRITE (6,1) X(1),Y(1),Z(1)                                         M1101820
      WRITE (6,72)                                                       M1101830
   72 FORMAT ('+1 2 3')                                                  M1101840
      CALL SUBMAT(C1,C2,C3,K1,C4,C5,C6,K2,C7,C8,C9,K3)                   M1101850
      IF (Z(1).GE.5000.D0) GO TO 40                                      M1101860
      Z(1) = Z(1) +  50.D0                                               M1101870
      GO TO 30                                                           M1101880
   40 CONTINUE                                                           M1101890
      STOP                                                               M1101900
      END                                                                M1101910
      SUBROUTINE SUBMAT(AX,AY,AZ,AA,BX,BY,BZ,BB,CX,CY,CZ,CC)             M1101920
```

```
      REAL*8 AX,AY,AZ,AA,BX,BY,BZ,BB,CX,CY,CZ,CC,XX,YY,ZZ,DD            M1101930
    7 FORMAT ('+',26X,'MATRIX CANNOT BE CALCULATED.')                   M1101940
      DD = AX*((BY*CZ)-(BZ*CY)) - BX*((AY*CZ)-(AZ*CY)) + CX*((AY*BZ)-(AZM1101950
     **BY))                                                             M1101960
      IF (DD.NE.C.D0) GO TO 98                                          M1101970
      WRITE (6,7)                                                       M1101980
      GO TO 99                                                          M1101990
   98 XX = (AA*((BY*CZ)-(BZ*CY)) - BB*((AY*CZ)-(AZ*CY)) + CC*((AY*BZ)-(AM1102000
     *Z*BY)))/DD                                                        M1102010
      YY = (AX*((BB*CZ)-(BZ*CC)) - BX*((AA*CZ)-(AZ*CC)) + CX*((AA*BZ)-(AM1102020
     *Z*BB)))/DD                                                        M1102030
      ZZ = (AX*((BY*CC)-(BB*CY)) - BX*((AY*CC)-(AA*CY)) + CX*((AY*BB)-(AM1102040
     *A*BY)))/DD                                                        M1102050
      WRITE (6,2) XX,YY,ZZ                                              M1102060
    2 FORMAT (10X,'CALCULATED',4X,3F10.2)                               M1102070
      WRITE (6,60)                                                      M1102080
   60 FORMAT ('+------------------------------------------------------') M1102090
   99 RETURN                                                            M1102100
      END                                                               M1102110
```

APPENDIX C

EXAMPLE OUTPUT OF APPENDIX B

An example of the output of the program in Appendix B is shown below. The total output for a single time step is presented.

```
SENSOR          SLANT          SENSED            CALCULATED
NUMBER          RANGE          SIGNAL              SIGNAL
   1           5050.0       0.6054D-04         0.6054E-04
   2           5060.1       0.1596D-03         0.1612E-03
   3           5050.2       0.2682D-04         0.2683E-04
   4           5099.0       0.3940D-04         0.4135E-04
   5           5050.2       0.9366D-04         0.9368E-04

             EQUATION              TRUE                     APPROXIMATE
               S2-S3            0.1327D-03                  0.1343E-03
               S2-S5            0.6590D-04                  0.6748E-04
               S3-S1            -.3372D-04                  -.3371E-04
               S5-S1            0.3312D-04                  0.3314E-04
               S3-S4            -.1258D-04                  -.1452E-04
               S5-S4            0.5426D-04                  0.5233E-04

EQNS                            X           Y          Z
USED                           POS         POS        POS
1 2 3      ACTUAL              0.0     -1000.00   -4950.00
_________CALCULATED_________-60.98___-870.83___-4739.97
```

The first five rows of data above are for the appropriate "sensor" as indicated, the slant range from that sensor to the target, the received signal as generated by equation 2.5 and the signal that would be generated using the approximation described in Chapter 3 to linearize the problem. The next six rows show the results of taking the differences between the received and calculated (for comparison purposes only) signals, listed under the headings of TRUE and APPROXIMATE, respectively. The final two lines show the actual coordinates of the platform relative to the target (the coordinate transformation has not been done at this point) as compared to the calculated position. These are the points used by the program in Appendix A to plot the graphs after the coordinate transformation is completed.

The second section of the output is shown below.

```
EQUATION     CONSTANT          X              Y              Z
 NUMBER                   COEFFICIENT    COEFFICIENT    COEFFICIENT
   1        0.50D+03      -0.30D+02      -0.92D+01       0.20D+01
   2       -0.38D+03       0.14D+00       0.13D+00       0.54D-01
   3       -0.86D+02      -0.21D+01      -0.19D+02       0.35D+01


  RATIO
   1/2     -0.13E+01      -0.22E+03      -0.71E+02       0.36E+02
   1/3     -0.58E+01       0.14E+02       0.49E+00       0.56E+00
   2/3      0.44E+01      -0.65E-01      -0.69E-02       0.15E-01
```

The first three lines of the output represent the coefficients of the equations 3.6 - 3.8 in the form of CONSTANT = X COEFF(x) + Y COEFF(y) + Z COEFF(z). In order to determine the linear independence of these equations, the ratios of the coefficients are taken in pairs. Since it may be seen from the last three lines of the output that the ratios of the constants and coefficients differ significantly for all three equations, it can be concluded that the equations are linearly independent. This holds true throughout any given run, which may be verified by printing out all time steps for that run.

APPENDIX D

CRAMER'S RULE

A. GENERAL

Cramer's Rule, a method for solving systems of linear equations in two or more unknowns, states that each unknown can be expressed as the ratio of two determinates. The general form of a solution is shown below.

$$Ax+By+Cz=O$$
$$Ex+Fy+Gz=H$$
$$Ix+Jy+Kz=L$$

$$D = \begin{vmatrix} A & B & C \\ E & F & G \\ I & J & K \end{vmatrix}$$

$$x = \begin{vmatrix} O & B & C \\ H & F & G \\ L & J & K \end{vmatrix} / D$$

$$y = \begin{vmatrix} A & O & C \\ E & H & G \\ I & L & K \end{vmatrix} / D$$

$$z = \begin{vmatrix} A & B & O \\ E & F & H \\ I & J & L \end{vmatrix} / D$$

B. EXAMPLE

The following example was taken from [Ref. 4]. Solve:

$$3x-2y+2z=7$$
$$x+y+z=6$$
$$2x-y-2z=2$$

Let D be the denominator.

$$D = \begin{vmatrix} 3 & -2 & 2 \\ 1 & 1 & 1 \\ 2 & -1 & -2 \end{vmatrix} = -17$$

$$x = \frac{\begin{vmatrix} 7 & -2 & 2 \\ 6 & 1 & 1 \\ 2 & -1 & -2 \end{vmatrix}}{D} = \frac{-51}{-17} = 3$$

$$y = \frac{\begin{vmatrix} 3 & 7 & 2 \\ 1 & 6 & 1 \\ 2 & 2 & -2 \end{vmatrix}}{D} = \frac{-34}{-17} = 2$$

$$z = \frac{\begin{vmatrix} 3 & -2 & 7 \\ 1 & 1 & 6 \\ 2 & -1 & 2 \end{vmatrix}}{D} = \frac{-17}{-17} = 1$$

CHECK:

$3x-2y+2z=7$

$3(3)-2(2)+2(1)=7$

$9-4+2=7$

$7=7$

$x+y+z=6$

$3+2+1+6$

$6=6$

$2x-y-2z=2$

$2(3)-2-2(1)=2$

$6-2-2=2$

$2=2$

C. SUBROUTINE "CRAMER'S RULE" CHECK

Running the values in Section B through the subroutines of the programs in Appendix A and Appendix B yielded the same results as the example.

LIST OF REFERENCES

1. Wynn,W.M., Frahm,C.P., Carroll,P.J., Clark,R.H., Wellhoner,J., Wynn,M.J, Advanced superconductive gradiometer/magnetometer arrays and a novel signal processing technique, p.701-707, IEEE Trans. Magnetics, MAG-11, 1975

2. Fromm,W.E, in Advances in Electronics, V.4, edited by Marton,L., p.260, Academic Press, 1952

3. Reitz,J.R., Milford,F.J., Foundations of Electromagnetic Theory, p.164, Addison-Wesley Publishing Co., Inc., 1967

4. Shapiro,M.S., Mathematics Encyclopedia, p.49, Doubleday & Company, Inc., 1977

5. Anderson,J.E., NADC Report NADC-EL-47-50, Magnetic Airborne Detection Frequency Responses, 1949

BIBLIOGRAPHY

BOEING Aerospace Company Proprietary Data, NAVY/BOEING Development of a Dual MAD Localization System, 8 November 1974

BOEING Aerospace Company, Dual MAD Processing Study, Task 1 Report, Advanced MEG Program, 1 November 1976

BOEING Aerospace Company, Dual MAD Processing Study, Task 2 Report, Techniques Analysis, 1 December 1976

National Defense Research Committee Division 6 Technical Report, Magnetic Airborne Detection Program, V.5, 1946

Naval Air Development Center Technical Memorandum NADC-3012:ARO, Extrema, symmetry, and zeroes of MAD signals and their application to MAD localization, 20 October 1982

TEXAS INSTRUMENTS INCORPORATED, Analysis of Transverse Gradiometer MAD using Dual Scalar Magnetometers, Part I: Detection Model and Performance, McGregor,D., July 1976

TEXAS INSTRUMENTS INCORPORATED, Analysis of Transverse Gradiometer MAD using Dual Scalar Magnetometers, Part II: Tactical Parameter Extraction, McGregor,D., January 1978

TEXAS INSTRUMENTS INCORPORATED, Flight Test Plan Dual MAD, McGregor,D., undated

INITIAL DISTRIBUTION LIST

		No. Copies
1.	Fleet Tactical Library P.O.Box 1042 Silver Springs, Md. 20910	2
2.	Defense Technical Information Center Cameron Station Alexandria, Virginia 22314	2
3.	Library, Code 0142 Naval Postgraduate School Monterey, Ca. 93943	2
4.	Dr. R. N. Forrest, Code 71 Chairman ASW Academic Group Naval Postgraduate School Monterey, Ca. 93943	2
5.	Dr. Otto Heinz, Code 61Hz Department of Physics Naval Postgraduate School Monterey, Ca. 93943	2
6.	Dr. Andrew R. Ochadlick, Jr., Code 610c Department of Physics Naval Postgraduate School Monterey, Ca. 93943	2
7.	LCDR C. Burmaster, USN, Code 61Zr Department of Physics Naval Postgraduate School Monterey, Ca. 93943	1
8.	LT Wolf H. Bock, USN ASWOC Sigonella FPO New York, N.Y. 09523	2
9.	Mr. John Shannon Naval Air Development Center (Code 3012) Warminster, Pa. 18974	1
10.	Mr. Edward Yannuzzi Naval Air Development Center (Code 30) Warminster, Pa. 18974	1
11.	Chief of Naval Research Department of the Navy Attn: John G. Heacock (Code 425 GG) 800 North Quincy Street Arlington, Va. 22217	1
12.	Mr. William Andahazy Naval Ship Research & Development Center Annapolis Laboratory Annapolis, Md. 21402	1

Thesis

202560
Thesis
B58895 Bock
c.1 A linear approximation of the source position using multiple MAD.

7 SEP 90 37053

202560
Thesis
B58895 Bock
c.1 A linear approximation of the source position using multiple MAD.

DUDLEY KNOX LIBRARY
3 2768 00018676 1